Article

Asynchronous Chirp Slope Keying for Underwater Acoustic Communication

Dominik Jan Schott [1,†,*], Andrea Gabbrielli [2], Wenxin Xiong [2], Georg Fischer [3], Fabian Höflinger [1,3], Johannes Wendeberg [2], Christian Schindelhauer [2], Stefan Johann Rupitsch [1]

[1] Dep. of Microsystems Engin. (IMTEK), University of Freiburg, Germany
[2] Dep. of Computer Sci. (IIF), University of Freiburg, Germany
[3] Fraunhofer EMI, Efringen-Kirchen, Germany
* Correspondence: dominik.jan.schott@imtek.uni-freiburg.de
† Current address: Georges-Koehler-Allee 106, 79110 Freiburg, Germany

Abstract: We propose an asynchronous acoustic chirp slope keying to map bit sequences on single or multiple bands without preamble or error correction coding on the physical layer. Details of the implementation are disclosed and discussed, the performance verified on laboratory scale in a pool measurement, as well as simulated for a channel containing Rayleigh fading and Additive White Gaussian Noise. For time-bandwidth products of 50 in single band mode, a raw data rate of 100 bit/s is simulated to achieve bit error rates below 0.001 for signal-to-noise ratios above -6 dB. In dual-band mode and a data rate of 200 bit/s, this bit error level was achieved for signal-to-noise ratios above 0 dB for time-bandwidth product of 25. The packet error rates follow this behavior with an offset of 1 dB.

Keywords: Underwater Communication; Wireless Communication; Acoustic Communication; Ultrasound Acoustics; Digital Signal Processing; Chirp Modulation; Chirp Slope Keying; Chirp Spread Spectrum;

1. Introduction

Shallow water still challenges communication attempts after over a hundred years of research [1–3] due to the strongly selective frequency fading, high phase noise and fast echoes and for moving nodes, due to a strong Doppler effect [4–9]. The shallower the channel, the more pronounced this inhibitions become. Previous investigations into the field of acoustic underwater communications have shown promising results [10–12], but concentrated on deeper bodies of water over longer distances of several kilometer in audible or sub-audible frequencies[13,14]. While high-bandwidth communication with large spectral efficiencies will be prone to upset the natural habitat if performed in the audible range of the maritime fauna [15,16], narrow-band methods are vulnerable the fading effects discussed before. The ongoing interest is a result of the strong attenuation of radio signals underwater and the long distances that require to be covered. Application examples, where underwater acoustic communication is crucial are diver tracking [17,18], robot/AUV telemetry [19–22], and underwater sensor networks [23,24].

1.1. Related Work

The general idea to sweep the frequency of a carrier to transmit information may be as old as (breathing) life on earth, but the first modern record we found is E. Hüttmann's patent for a distance measurement method from 1940 [25]. The idea to use chirps for modulating a signal, e.g., as Chirp Slope Keying (CSK) is often attributed to M. R. Winkler's work in the early 1960s [26], and also sometimes refered to as Linear Frequency Modulation or Linear FM [27]. After the application in satellite radio transmission during the Cold War era [28,29], the interest in chirp modulations mostly vanished due to the low achievable data rates. For radio communications, this changed in 2013 with the establishment of the LoRa protocol as a low-power long range radio modulation for small consumer applications [30,31]. Chirp modulation of acoustic signals under water have gathered attention since the early 2000s [32], and is researched continuously by groups around the world ever since [18,33–37]. While the achievable data rate is severely limited compared to narrow-band schemes [5], CSK will be especially of interest, when the data amount is low and the channel inhibitions are strong. The performance of our approach in this work has partially been reported in [38]. The scope of this contribution is to report on the details of our system's inner structure and underlying algorithms. This investigation zooms in on the modulation and demodulation part of typical basic elements of digital communication systems [39], hence, expects coded input and will return still coded output. Consequently, additional forward error correction coding will improve the estimation of the overall system [40], but is not part of this investigation.

2. Materials and Methods

2.1. Basic System Structure

Our system is divided into functional blocks with the modulation and demodulation in focus, as shown in Fig. 1; more details about the structure inside those two blocks is discussed in sec. 2.2 and sec. 2.4.

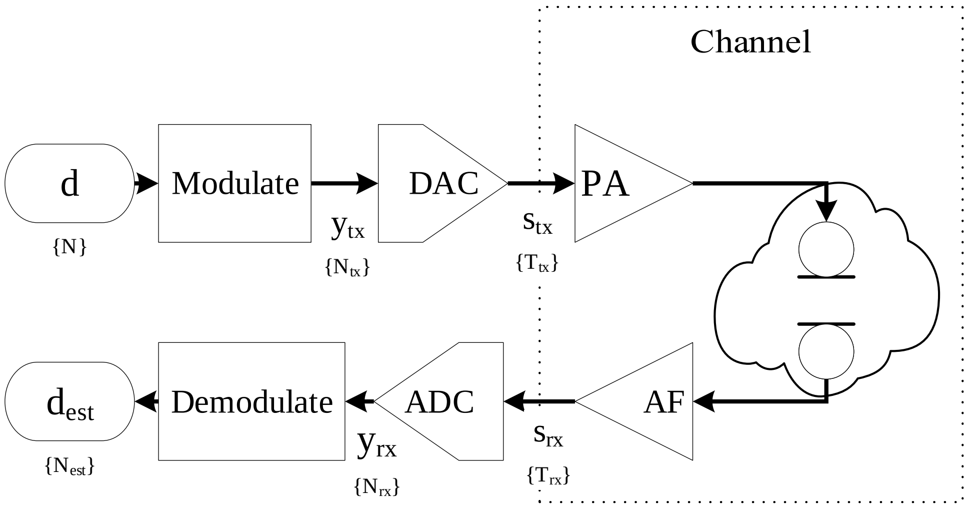


Figure 1. Flow diagram of the basic communication chain: The data d is modulated, amplified before transmission, filtered and amplified at reception, and demodulated as d_{est}. The entire analog domain is regarded as part of the communication channel.

The data d of length N is modulated into the digital sequence of linear up and down chirps y_{tx}, now of length N_{tx}, as illustrated in Fig. 1. The DAC then converts this into the the output s_{tx}, which is an analog continuous real-valued signal of length T_{tx} that is boosted by a power amplifier (PA) before it is turned into an acoustic wave by a piezoelectric transducer.

The received signal is bandpass-filtered and amplified by an analog active filter (AF) into the continuous real-valued and band-limited received signal s_{rx} of length T_{rx}. The ADC samples the received signal into the sequence y_{rx} of length N_{rx}. The demodulation step estimates the originally sent sequence as d_{est} given a small set of prior information about the original signal, e.g., the ideal chirp parameters.

We consider exclusively time discrete signals throughout this work, sampled at points

$$n = \lfloor t \cdot f_{\mathrm{s}} \rfloor \in \mathbb{Z}, \tag{1}$$

where t is the time of observation and f_s the sampling frequency. For simplicity we assume the sampling frequency of transmitter and receiver to be equal, save an oscillator frequency mismatch of Δf_s and phase offset of $\Delta\phi_s$. In practical application, this is not required, but only the parameters that describe the used waveform sufficiently.

2.2. *Modulation*

Before transmission the data d is mapped onto the slope sign of chirps and up-converted into the transmission bands, see Fig. 2.

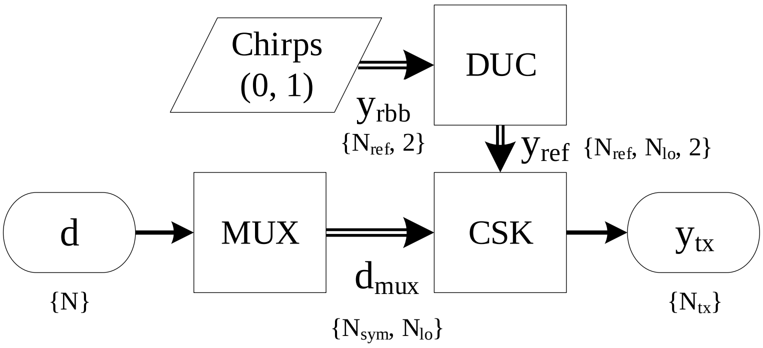


Figure 2. Modulator block in detail: The data input d is multiplexed onto N_{lo} sub-bands (MUX) and modulated by the up-converted chirped symbols from the *Digital Up-Converter* (DUC). The transmission sequence y_{tx} is assembled by the *Chirp Slope Keying* (CSK) block, already in the transmission band. Simple arrow lines indicate vectors, double lines arrays.

2.2.1. Linear Chirp Creation

Initially, a reference chirp y_{rbb} is generated through

$$y_{\mathrm{rbb}}[n] = \begin{cases} w[n]\cdot\sin(\varphi[n]),\text{ with,} & \text{for } 0 < n \le N_{\mathrm{ref}} \\ 0, & \text{else.} \end{cases} \tag{2}$$

For simplicity of calculation we normalize[1] the angular frequencies

$$\begin{aligned} \omega_0 &= 2\pi f_0/f_s,\text{ and} \\ \omega_1 &= 2\pi f_1/f_s \end{aligned} \tag{3}$$

with the start frequency f_0 and the stop frequency f_1. This allows the definition of the argument $\varphi[n]$ for a linear chirp as

$$\omega[n] = \omega_0 + n\frac{\omega_1-\omega_0}{N_{\mathrm{ref}}}, \tag{4}$$

and therefore

$$\varphi[n] = \varphi_0 + \int_0^n \omega[\nu]\mathrm{d}\nu. \tag{5}$$

With (4) and (5), we can calculate the instantaneous phase for a linear sinusoidal chirp according to

$$\varphi[n] = \varphi_0 + n\cdot\omega_0 + \tfrac{1}{2}n^2\cdot\frac{\omega_1-\omega_0}{N_{\mathrm{ref}}}. \tag{6}$$

The modulation will become clearer if we substitute

$$\omega_c = \frac{\omega_1+\omega_0}{2}, \tag{7}$$

---

[1] Note that implementations in MATLAB often use the Nyquist frequency $f_{\mathrm{nyq}} = f_s/2$ for normalization instead.

$$B_\omega = |\omega_1 - \omega_0|, \tag{8}$$

and introduce

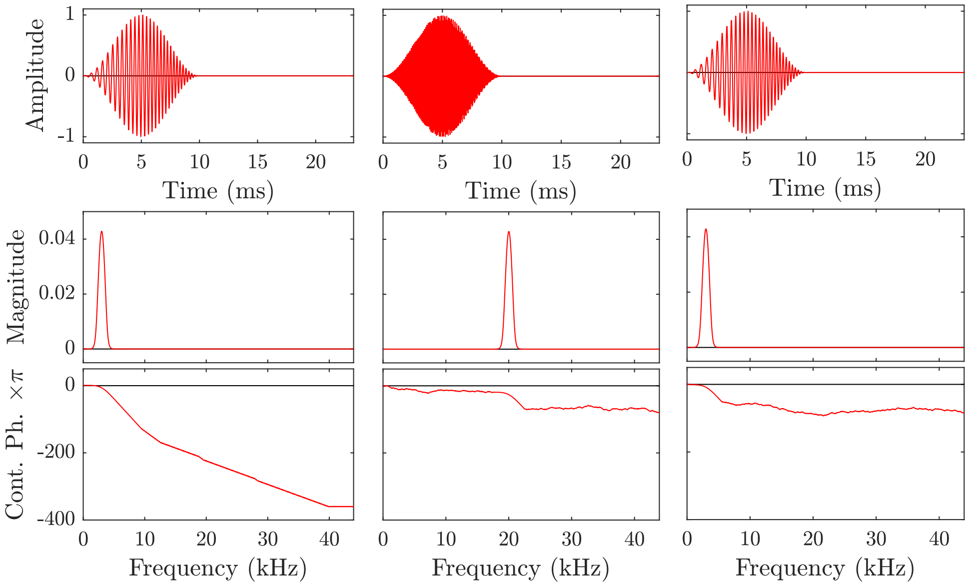
$$\zeta = \operatorname{sign}(\omega_1 - \omega_0). \tag{9}$$

The instantaneous phase of the chirp from (6) then takes the form of

$$\varphi[n] = \varphi_0 + n \cdot \omega_c + \zeta \frac{B_\omega}{2} n \left(1 + \frac{n}{N_{\text{ref}}}\right), \tag{10}$$

where each bit of the data is mapped onto the sign ζ.

The resulting chirp sequence is generated in the base-band frequencies according to (2) and up-converted to each channel. In Fig.3a an example for a base-band chirp is shown and the result of the up-conversion in Fig.3b.



(a) **(b)** **(c)**

Figure 3. Resampling example for a linear chirp with $f_c = 3\,\text{kHz}$, $B_f = 2.5\,\text{kHz}$, and $T = 10\,\text{ms}$. **Left:** Base band signal y_{tb} at the transmitter, **middle:** Transmission band y_{ib}, **right:** Base band signal y_{bb} at the receiver. The transmission band's center frequency is at 20 kHz for clarity. In the experiments, the transmission band is around 62.5 kHz.

For a fixed transmission channel communication, the up-conversion through a DUC can be omitted and the reference chirps directly be calculated in the transmission band, but for the sake of flexibility, we added the up-conversion as a separate block. The resulting single chirp sequences y_{ref} for all N_{lo} transmission channels can be stored permanently and only requires to be recalculated, if the parameters, e.g., sampling frequency f_s, chirp length N_{ref}, side-band center frequency f_{ch} or bandwidth B change. While intuitively both chirp slope sequences may be pre-generated, we omit this redundancy on implementation as the inverse slope sign is equivalent to a time reversal of the entire chirp sequence.

2.2.2. Shaping

The amplitude shaping window $w[n]$ restricts the sequence to be non-zero in the interval between 0 and N_{ref} only. While this can be achieved through different window functions, the tapered cosine, i.e., Tukey window is used in this work, because it can be varied easily between the rectangular, i.e., Dirichlet window and a sine, i.e., Hann window, by changing the single tuning factor a_t to 0 or 1, respectively [41]. This sets the

doi:10.20944/preprints202103.0750.v1

main lobe width between $4\pi/N_{\text{ref}}$ and $8\pi/N_{\text{ref}}$, as well as the peak sidelobe between −13 dB and −31 dB [42]. We define the Tukey window as

$$w[n] = \begin{cases} \frac{1}{2}\left[1 - \cos\left(\frac{\pi}{N_{\text{tk}}} n\right)\right], & \text{for } 0 < n \leq N_{\text{tk}} \\ 1, & \text{for } N_{\text{tk}} < n \leq (N_{\text{ref}} - N_{\text{tk}}) \\ \frac{1}{2}\left[1 - \cos\left(\frac{\pi}{N_{\text{tk}}}(n - (N_{\text{ref}} - N_{\text{tk}}))\right)\right], & \text{for } (N_{\text{ref}} - N_{\text{tk}}) < n \leq N_{\text{ref}} \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where the threshold of the taper is set by

$$N_{\text{tk}} = \frac{a_{\text{t}} N_{\text{ref}}}{2}.$$

The amplitude shape of the chirp can thus be adapted to the channel, depending on the application, i.e., if a narrow autocorrelation peak width is required for spatial distinction of two close reverberations or wide smooth peaks are desired for more robust communication. A simplified comparison of a selection of window functions for a fast echo is shown in Fig. 4 and Fig. 5.

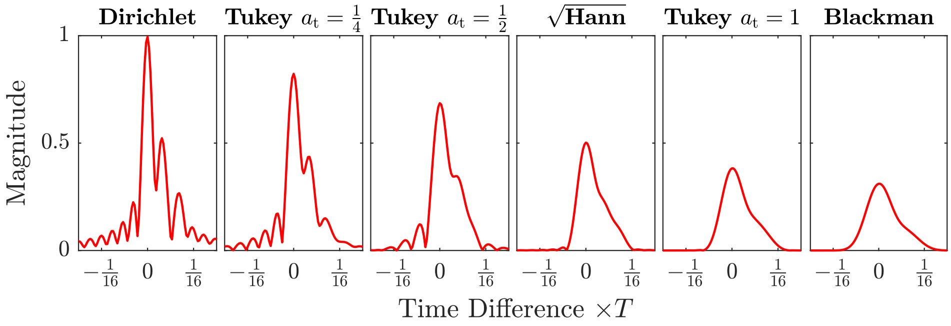


Figure 4. Autocorrelation magnitude comparison of a selection of shaping window functions. All magnitudes are normalized by the Dirichlet shaped chirp power for comparison. Gaussian noise was added to a $SNR = 0$ dB, as well as two echoes at $n = 9$ and $n = 19$.

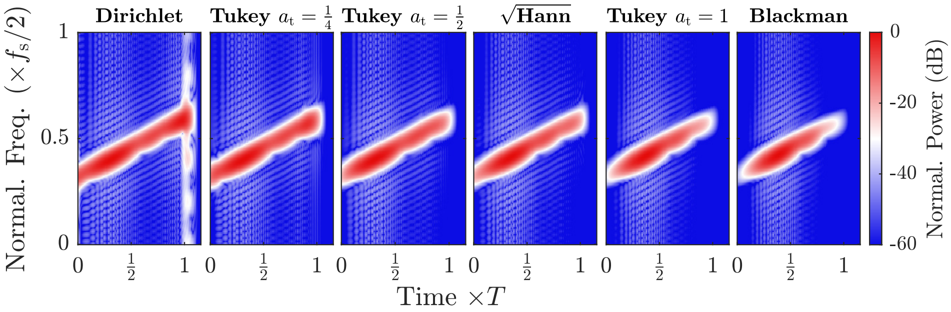


Figure 5. Simulated spectrograms of the autocorrelations of a selection of shaping window functions. All magnitudes are normalized by the Dirichlet shaped chirp power for comparison. Gaussian noise was added to a signal-to-noise ratio of 0 dB, as well as two echoes at $n = 9$ and $n = 19$.

2.2.3. Input Multiplexing

When data d of length N is submitted for transmission, the 1-D binary sequence is first multiplexed onto N_{lo} channels through

$$\begin{aligned} d_{\mathrm{mux}}[k,\, l] &= d[n], \quad \text{where} \\ n &= N_{\mathrm{lo}} \cdot (k-1) + l, \\ k &\in \left[1,\, N_{\mathrm{sym}}\right], \\ l &\in [1,\, N_{\mathrm{lo}}]. \end{aligned} \tag{12}$$

The multiplexed sequence length N_{sym} is

$$N_{\mathrm{sym}} = \left\lfloor \frac{N}{N_{\mathrm{lo}}} \right\rfloor, \tag{13}$$

therefore N needs to be an integer multiple of N_{lo}. To assert this, we use a simple zero-padding algorithm. If bits are added, they will remain in the data on reception and need to be removed on a higher level later on. This can be avoided, by matching the data length to the desired number of channels in advance, but for a more flexible and general approach, we implement the zero-padding approach and truncate to byte-sizes of 8.

2.2.4. Chirp Slope Keying

A fast way to modulate the binary sequence is to have a simple decision block, that will output a chirp sequence of either upward or downward slope according to the binary value at the input.The segments of the ouput sequence y_{tx} are assembled by filling each interval of length N_{ref} with the normalized sum of the superposed channels' sequences to

$$y_{\mathrm{tx}}[n] = \frac{1}{N_{\mathrm{lo}}} \sum_{l}^{N_{\mathrm{lo}}} y_{\mathrm{ref}}[n,\, l,\, d_{\mathrm{mux}}[k,\, l]]. \tag{14}$$

The output sequence is transferred into an analog signal s_{tx}, e.g., by a DAC, amplified by the PA and transmitted. Alternatively, this modulation can be calculated by zero-padding the multiplexed bit sequences d_{mux} by length N_{ref} and convolving the resulting sequence with the reference signal y_{ref}, but this approach is neither efficient in memory usage, nor the calculation steps required [43].

2.3. Channel Model

We adapted the simulation approach from [33] to estimate the performance of the modulation and demodulation modules in a controlled fashion. The channel model includes simplified Rayleigh fading that multiplies the signal amplitude by the magnitude of two independent, but identically distributed random processes

$$\begin{aligned} A_{\mathrm{r}} &= |\sigma_r \cdot (\mathrm{randn}\{N_{\mathrm{rx}}\} + \mathrm{i} \cdot \mathrm{randn}\{N_{\mathrm{rx}}\})|, \quad \text{where} \\ \mathrm{i} &= \sqrt{-1}, \end{aligned} \tag{15}$$

with a distribution parameter $\sigma_{\mathrm{r}} = 1$. The random sequences are generated through the *randn* function of MATLAB that generates a normally distributed random value. An Additive White Gaussian Noise (AWGN) in the form of

$$\epsilon_{\mathrm{n}} = \sigma_{\mathrm{g}} \cdot \mathrm{randn}\{N_{\mathrm{rx}}\} \tag{16}$$

is used to model the thermal noise of the receiver.

For simulations, we assume the receiver samples a combination (15) and (16) with the transmitted signal, at a random packet reception time offset n_τ, without additional reverberations from multiple paths

$$y_{\mathrm{rx}}[n] = A_{\mathrm{r}} \cdot y_{\mathrm{tx}}[n + n_\tau] + \epsilon_{\mathrm{n}}. \tag{17}$$

2.4. Demodulation

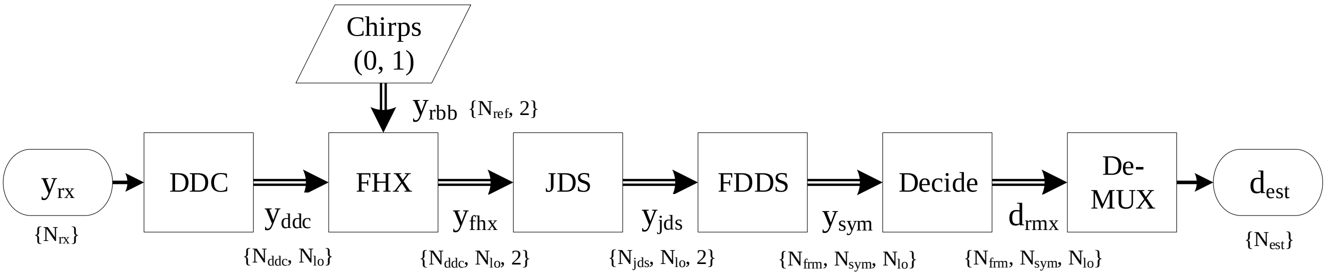


Figure 6. Demodulator block in detail: The received sequence y_{rx} is processed by *Digital Down-Coverter* (DDC), *Fast Hilbert Cross-Correlator* (FHX), *Join & Downsample* (JDS), *Frame Detect & DownSample* (FDDS), a binary *Decide* block, and a reverting multiplexer (De-MUX) generating the estimated data output d_{est}.

The digitized signal y_{rx} is translated into the baseband for each of the N_{lo} channels in the *Digital Down-Converter* (DDC) block, as shown in Fig.6. The *Fast Hilbert Cross-Correlator* (FHX) block compresses the signal further into arrays y_{fhx} for additional dimensions for each of the reference chirps of both slope signs. The block *Join & Downsample* (JDS) attempts coherent addition and subtraction of the 2 signal arrays for each channel. The resulting sum and difference signals in in y_{jds} are analyzed by the *Frame Detect & Downsample* (FDDS) block and the input signal divided into separate frames y_{sym}, now at symbol rate. The final decision block translates the symbols into binary values d and estimates the demodulation performance. Each block is described below in detail.

2.4.1. Digital Down-Converter

Before the signal is fed into the resource intensive compression algorithm, we exploit the bandlimited nature of the signal and bring it down into the baseband, by calculating

$$\begin{aligned} y_{\mathrm{tb}}[n, c] &= \mathrm{BPF}\{y_{\mathrm{rx}}[n]\}, \\ y_{\mathrm{ib}}[n, c] &= y_{\mathrm{tb}}[n] \cdot y_{\mathrm{lo}}, \\ y_{\mathrm{bb}}[n, c] &= \mathrm{LPF}\{y_{\mathrm{ib}}[n, c]\}, \end{aligned} \tag{18}$$

where the functions LPF denotes an arbitrary lowpass filter, and BPF any suitable bandpass filter. The signal content outside of the band is suppressed by the analog bandpass filter of the receivers signal conditioning before the sampling. This is especially important for undersampling a signal to limit the aliasing effect of noise. To achieve the downconversion we first multiply the bandpass filtered raw signal y_{tb} of each transmission band with sine waves y_{lo} of frequency f_{lo} to create the intermediate signal y_{ib}. This operation shifts the content of each of the N_{lo} channels into the baseband, where a lowpass filter removes the higher harmonics and produces the baseband sequence y_{bb}. In doing so, the memory consumption increases by the number of channels N_{lo}, a one-dimensional real-valued input sequence of $[N_{\mathrm{rx}}, 1]$ gets mapped onto an $[N_{\mathrm{rx}}, N_{\mathrm{lo}}]$ output array. The sequence can be truncated in frequency domain to an interval around the center, since most of the frequency bands ideally contain no information about the signal and one loses only information about noise and interference. We effectively resample the sequence to

$$\begin{aligned} y_{\mathrm{ddc}} &= \mathrm{resample}\{y_{\mathrm{bb}}, f_{\mathrm{s}}, f_{\mathrm{s1}}\}, \text{ where} \\ f_{\mathrm{s1}} &= \frac{f_{\mathrm{s}}}{N_{\mathrm{res1}}}. \end{aligned} \tag{19}$$

As we use the single sideband approach, a minimal interval is limited by the center frequency

$$f_{\mathrm{c}} = \tfrac{1}{2}(f_1 + f_0) \tag{20}$$

and half the bandwidth

$$B_{\mathrm{f}} = (f_1 - f_0). \tag{21}$$

Assuming a sampling rate of, e.g., $f_{\mathrm{s}} = 88\,\mathrm{kHz}$, a bandwidth of $B_{\mathrm{f}} = 2.5\,\mathrm{kHz}$ and a sub-band center frequency of $f_{\mathrm{c}} = 3.0\,\mathrm{kHz}$ as used in the dual-band case, the minimal one-sided base band is

$$B_{\mathrm{fbbm}} = f_{\mathrm{c}} + \tfrac{1}{2}B_{\mathrm{f}}, \tag{22}$$

which is for the given example $B_{\mathrm{fbbm}} = 4.25\,\mathrm{kHz}$. Considering the original sample bandwidth and unchanged frequency bin width, the computation is reduced to $B_{\mathrm{fbbm}}/\frac{1}{2}f_s$, here by about 90 % at most. The minimal interval truncation also removes information about the noise, so a trade-off is feasible that implements a larger interval of several bandwidths. Moreover, the whole band-shifting and resampling can effectively be done in the frequency domain with a shift and truncate operation. An example for this operation in shown in Fig.3b for the transmission band sequence and Fig.3c for the down-sampled result.

2.4.2. Pulse Compression by Fast Hilbert Cross-Correlation

If time and magnitude of a received chirp are of interest, the calculation of the analytic signal after pulse compression through a matched filter is convenient. Hence, the next signal processing step is to convolve (operator $\circledast$) the received signal with the matched filter for both chirp slope signs

$$\begin{aligned} y_{\mathrm{mf}\uparrow} &= y_{\mathrm{ddc}} \circledast y_{\mathrm{rbb}\uparrow}, \\ y_{\mathrm{mf}\downarrow} &= y_{\mathrm{ddc}} \circledast y_{\mathrm{rbb}\downarrow}, \end{aligned} \tag{23}$$

This increases the memory allocation to $[N_{\mathrm{rx}}, N_{\mathrm{lo}}, 2]$ samples, as the downsampled sequences are compressed by both, up and down chirps. In case more different chirps are used, this increases the added dimension accordingly. The compressed pulse's envelope is then calculated as the analytic signal through the norm of the signal and its Hilbert transform

$$y_{\mathrm{fhx}} = \sqrt{y_{\mathrm{mf}}^2 + \mathcal{H}\{y_{\mathrm{mf}}\}^2}. \tag{24}$$

The calculations, both, matched filtering and envelope extraction, are performed in the frequency domain for convenience. After the Fourier transformation of the raw signal, we perform a bin-wise multiplication against the complex conjugated reference signals to obtain the compressed signals for both up and down chirps.

2.4.3. Join & Downsample

Frame detection and symbol decision require information about the compressed pulse peak positions in time, which are difficult to establish in one matched filter branch, e.g., only the up chirp compression result, as there may be no peaks present, if the signal hypothetically only consists of down chirps. The *Join & Downsample* block first resamples the sequence to an integer fraction by N_{res2} to the sample rate

$$\begin{aligned} y_{\text{res2}} &= \text{resample}\{y_{\text{fhx}}, f_{\text{s1}}, f_{\text{s2}}\}, \text{ where} \\ f_{\text{s2}} &= \frac{f_{\text{s1}}}{N_{\text{res2}}}, \\ N_{\text{jds}} &= \left\lfloor \frac{N_{\text{fhx}}}{N_{\text{res2}}} \right\rfloor, \end{aligned} \tag{25}$$

then creates the sum and difference

$$\begin{aligned} y_{\text{sum}}[n, c] &= y_{\text{fhx}\uparrow}[n, c] + y_{\text{fhx}\downarrow}[n, c], \\ y_{\text{dif}}[n, c] &= y_{\text{fhx}\uparrow}[n, c] - y_{\text{fhx}\downarrow}[n, c]. \end{aligned} \tag{26}$$

This operation requires coherence, since a phase difference between the up and down chirp compressed sequences leads to sub-optimal symbol detection. This condition will be fulfilled only if no Doppler shift is present, so sender and receiver do not move relative to each other [44]. For this work, we exclusively considered stationary conditions. The sum and difference sequences are stored in a joint array y_{jds} of size $\left\{N_{\text{jds}}, N_{\text{lo}}, 2\right\}$.

2.4.4. Frame Detect & Downsample

The FDDS block first estimates the frame positions in half symbol space, then uses this information to estimate the symbol phase of each frame and downsample it to full symbol space. First, we assume a known symbol length N_{ch2} from the reference chirp sequence and estimate it simply to

$$N_{\text{ch2}} = \lfloor T \cdot f_{\text{s2}} \rfloor = \left\lfloor N_{\text{ref}} \cdot \frac{f_{\text{s2}}}{f_{\text{s1}}} \right\rfloor. \tag{27}$$

The mean magnitude of each of the M_H half symbol frames of length N_{H}, where

$$N_{\text{H}} = \left\lfloor \frac{N_{\text{ch2}}}{2} \right\rfloor, \tag{28}$$

is then calculated by only regarding the superposed pulses of both channels, which guarantees the presence of an autocorrelation peak in each symbol. Therefore, we calculate

$$\begin{aligned} y_{\text{ms}}[m] &= \sum_{n}^{N_{\text{H}}} y_{\text{sum}}[n + N_{\text{H}} \cdot (m-1)], \quad \text{where} \\ m &\in [1, M_{\text{H}}], \\ M_{\text{H}} &= \left\lfloor \frac{N_{\text{jds}}}{N_{\text{H}}} \right\rfloor, \end{aligned} \tag{29}$$

which reduces strong magnitude fluctuations before the data frame detection and resamples the sequence to half symbol space. As the envelope detection is very sensitive to non-steady slopes, we apply an additional 10^{th} order lowpass filter

$$y_{\text{mLP}} = \text{LPF}\{y_{\text{ms}}\}, \tag{30}$$

with an estimated cutoff frequency

$$\begin{aligned} \omega_{\text{MLP}} &= \frac{m_{\text{MLP}}}{M_{\text{H}}}, \quad \text{where} \\ m_{\text{MLP}} &= \arg\max\{|\text{FFT}\{y_{\text{ms}}\}|\}. \end{aligned} \tag{31}$$

The frame detection algorithm has two parts. Initally, a threshold is calculated for the whole received sequence of each channel, then a state machine iterates through it and extracts frame start and end times. We estimate the threshold y_{th} by a simple clustering, that first calculates the mean amplitude of the lowpass filtered half symbol magnitude

$$\bar{y}_{\mathrm{mLP}} = \mathrm{mean}\{y_{\mathrm{mLP}}\}, \tag{32}$$

then calculates the cluster means for both sides of the mean level,

$$\begin{aligned} \bar{y}_{\mathrm{mS}} &= \mathrm{mean}\{y_{\mathrm{mLP}}[y_{\mathrm{mLP}} > \bar{y}_{\mathrm{mLP}}]\}, \\ \bar{y}_{\mathrm{mN}} &= \mathrm{mean}\{y_{\mathrm{mLP}}[y_{\mathrm{mLP}} < \bar{y}_{\mathrm{mLP}}]\}, \end{aligned} \tag{33}$$

where the lower mean value $\bar{y}_{\mathrm{mN}}$ is considered the noise level and the upper mean $\bar{y}_{\mathrm{mS}}$ the signal level. The threshold is then simply the arithmetic mean of those two levels

$$y_{\mathrm{mTh}} = \frac{(\bar{y}_{\mathrm{mS}} + \bar{y}_{\mathrm{mN}})}{2}. \tag{34}$$

Subsequently, the state machine iterates through the sequence y_{mLP} and records an upwards slope if there are M_{HL} of samples below the threshold y_{mTh} followed by M_{HH} samples above it. We set both intervals $M_{\mathrm{HL}} = M_{\mathrm{HH}} = 2$, limiting the miminal frame size to $M_{\mathrm{HL}} + M_{\mathrm{HH}} - 1 = 3$ samples. A state variable will keep track if the iteration is inside a frame and stores start index $m_0[p]$ and end index $m_1[p]$ of each p^{th} frame. The frame limits are then reconstructed in sample space through scaling the indice by M_{H},

$$n_0[p] = m_0[p] \cdot N_{\mathrm{H}}, \quad \mathrm{and} n_1[p] = m_1[p] \cdot N_{\mathrm{H}}. \tag{35}$$

The single frames in sample space are then defined as

$$\begin{aligned} y_{\mathrm{fSum}}[p, n] &= y_{\mathrm{sum}}[n_0[p] + n], \quad \text{where} \\ y_{\mathrm{fDif}}[p, n] &= y_{\mathrm{dif}}[n_0[p] + n], \quad \text{where} \\ n &\in [1, N_{\mathrm{frm}}[p]], \end{aligned} \tag{36}$$

where y_{sum} and y_{div} are the two sub-arrays of y_{jds} and include all N_{lo} channels as an additional dimension, respectively. The indexing of the channel dimensions has been omitted for ease of reading. The number of samples in each frame is

$$N_{\mathrm{frm}}[p] = n_1[p] - n_0[p]. \tag{37}$$

The last part of the block selects each data frame in the sample space, searches for the optimal sample offset n_{off} to maximize the symbol power and assembles a frame in symbol space accordingly. We assemble the power matrix for each frame p and each channel by iterating through the phase sample by sample

$$\begin{aligned} A_{\mathrm{y}}[p, n] &= \sum_{k}^{K[n]} (y_{\mathrm{fSum}}[p, n + (k-1) \cdot N_{\mathrm{ch2}}])^2, \quad \text{where} \\ K[n] &= \left\lfloor \frac{N_{\mathrm{jds}} - n}{N_{\mathrm{ch2}}} \right\rfloor. \end{aligned} \tag{38}$$

The optimal sample offset n_{off} is then estimated to

$$n_{\mathrm{off}}[p] = \arg\max_{n} \{A_{\mathrm{y}}[p, n]\}. \tag{39}$$

This we use to assemble the block's final 3-dimensional output sequences y_{sym}, that span the number of detected frames, in each of which the number of symbols, and a constant number of channels. Hence, occupy memory of size $[N_{\text{frm}}, N_{\text{sym}}, N_{\text{lo}}]$ as we decimate

$$y_{\text{sym}}[p, k] = y_{\text{fDif}}[p, n_{\text{off}}[p] + (k-1) \cdot N_{\text{ch2}}], \tag{40}$$

again the indexing for all channels is omitted.

2.4.5. Symbol Decision

The symbol decision iterates through each frame's symbol space difference sequence y_{sym} similarly to (32) to (34) of sec.2.4.4, by separating each frame in two clusters split by the mean symbol amplitude, and estimates the half distance between both clusters' means as a threshold y_{fTh} for symbol decision for each channel. The decision equation is, therefore, simply

$$d_{\text{rmx}}[k] = \begin{cases} 1, & \text{for } y_{\text{sym}}[k] > y_{\text{fTh}} \\ 0, & \text{otherwise,} \end{cases} \tag{41}$$

for the k^{th} symbol of each channel and frame.

2.4.6. De-Multiplexing

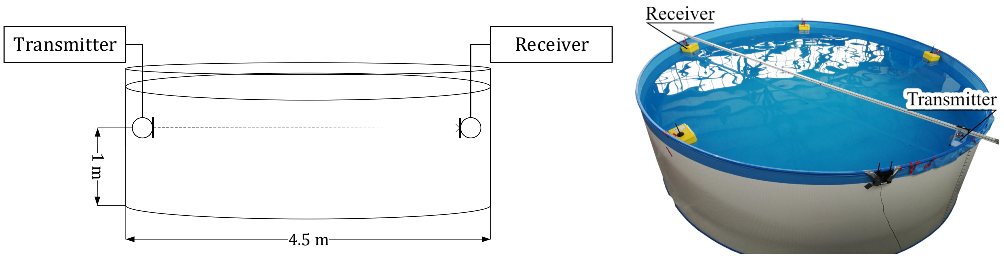
The last block of the demodulation chain re-assembles the N_{lo}-dimensional symbol sequences of each frame into a one-dimensional bit sequence. The length of the received bit sequence N_{est} is first truncated to multiples of 8, as the application is meant to send and receive data byte-wise, hence

$$N_{\text{est}} = 8 \cdot \left\lfloor \frac{N_{\text{lo}} \cdot N_{\text{sym}}}{8} \right\rfloor. \tag{42}$$

The data is then de-multiplexed by reshaping the sequences d_{rmx} with n in the range $[1, N_{\text{est}}]$ to

$$\begin{aligned} d_{\text{est}}[n] &= d_{\text{rmx}}[k, l], \quad \text{where} \\ k &= \frac{n}{N_{\text{lo}}}, \\ l &= n \bmod N_{\text{lo}}. \end{aligned} \tag{43}$$

2.5. Experimental Set-up



(a) Schematic set-up **(b)** Photo of the pool set-up

Figure 7. Schematic experimental set-up in for the acoustic transmission inside a water tank.

We conducted two experimental runs to verify our approach. One of a single band transmission, the other of a dual-band transmission. The experiments were performed in a steel-walled pool as shown in Fig.7, which was assembled temporarily inside a building. The transmitter and receiver hardware are a modified version of the indoor localization system [45], as we published before [17,38].

doi:10.20944/preprints202103.0750.v1

2.5.1. Frequency Band Considerations

Acoustic underwater communication influences the maritime habitat, hence system designs have to minimize the interruption of natural communication. As a simplified design rule, we regard that seals' hearing is sensitive to sound frequencies below 80 kHz, with already increased sound pressure level thresholds, i.e., decreased hearing sensitivity above 60 kHz [46]. While this differs for other species of sea mammals, which can create and react to sound of up to 150 kHz [47–49], we assume a similar decreased sensitivity of frequencies above 60 kHz. Additionally, the attenuation of acoustic underwater waves exceeds 20 dB km^{-1} for those frequencies, limiting the spatial sphere of influence further. A limiting factor for coastal applications is natural and artificial noise, e.g., from the surf and ship traffic, which we regard as Brownian noise decaying at about 18 dB per decade [16,50]. For our system, we limit the communication band therefore as in Table 1.

2.5.2. Experiment Parameters

Table 1. Transmission Band Parameters

Parameter	Value	Description
f_c	67.5 kHz	Center frequency
$\hat{B}$	5.0 kHz	Maximal available bandwidth
f_s	88.0 kHz	Receiver sampling frequency
N_{res}	4	Resampling factor after down-mixing
N_{res2}	2	Resampling factor after signal merge
f_{s1}	22 kHz	Sampling frequency after 1st downsampling
f_{s2}	11 kHz	Sampling frequency after 2nd downsampling

The sampling rate of our acquisition unit is limited to $f_s = 88$ kHz. As a result, the received signal is undersampled, i.e., the Nyquist frequeny is below the transmission band. While this mixing operation generally results in a leakage of signal power, the band-limited nature of the chirp sequences and the low noise environment limit this aliasing effect. This band-limitation is ensured by an additional analog bandpass-filter. The chirp parameters are listed in Table 2 for the single band and dual-band transmission.

Table 2. Experiment Waveform Parameters

Parameter	Value		Description
	Single	Dual	
N	96	64	Transmitted bits
N_{lo}	1	2	Number of sub-channels
N_{frm}	3	3	Number of packages sent
B	5.0 kHz	2.50 kHz	Bandwidth per channel
T	10 ms	10 ms	Length of a single chirp in time
f_c	67.5 kHz	[66.25, 68.75] kHz	Frequency offset to band center
$T \cdot B$	50	25	Time-bandwidth product

The symbol rate and occupied bandwidth of both transmissions is kept constant. Therefore, the single band signal has twice the time-bandwidth product compared to the dual-band one. This implicates the ratio of symbol energy to noise energy to double as well [51],

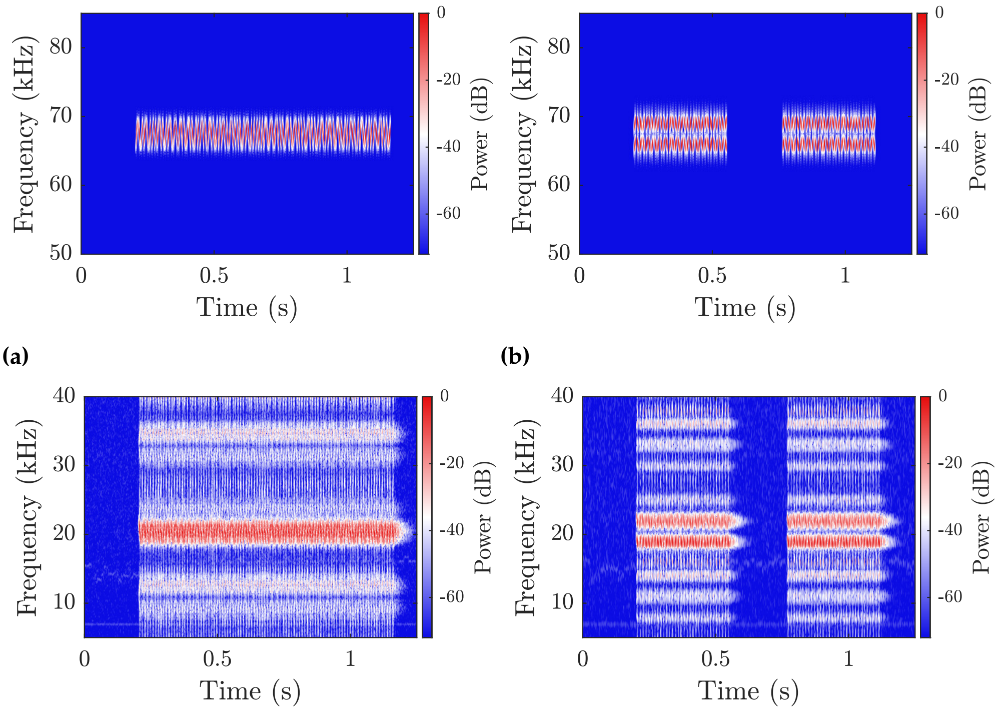
$$\frac{E_{sym}}{E_n} = T \cdot B \cdot \gamma, \tag{44}$$

where γ is the signal-to-noise ratio of the received signal. The expected data rate on the other hand is halved, as each symbol only contains half the bits.

3. Results

3.1. Channel Frequency Response

The transmitted and received signals are shown in Fig.8 as spectrograms over frequency and time. The undersampling introduces harmonic interference outside of the transmission band of the recorded signal, which are not physically present in the medium itself. Those phantom bands are removed on downsampling, by narrow bandpass filters.



(a) **(b)**

(c) **(d)**

Figure 8. Spectrograms of parts of the signal. **Top:** After the up-conversion in the transmitter; **Bottom:** Before down-conversion on the receiver side. **Left:** Plots of the single band communication; **Right:** Plots of the dual-band communication. Each package is transmitted three times in the experiment.

The power levels are more clearly visible in the averaged plots of Fig.9. The noise floor confirms the assumption of AWGN outside of the transmission, with an approximate SNR of 65 dB. The interference caused by the transmission itself raises the average power outside of the transmission band for about 30 dB.

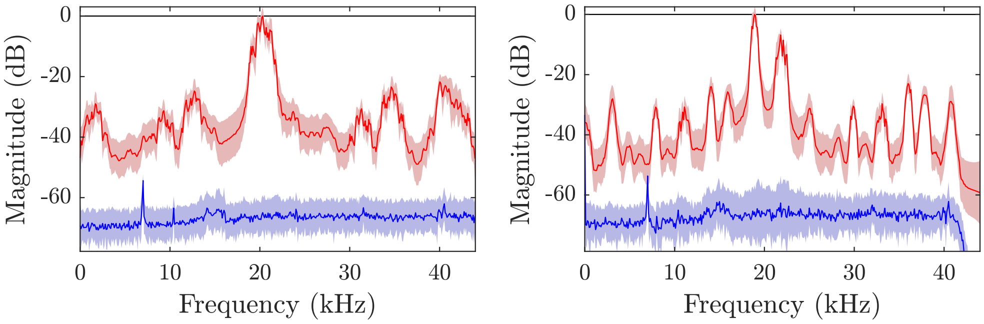
4. Bit Error Rate and Packet Error Rate Simulations

The bit error rate (BER) and packet error rate (PER) of the proposed algorithms are estimated through simulation for an idealized channel as described in sec.2.3. We define the BER in two ways: By comparing each bit in the order of demodulation through the exclusive or operation

$$r_{\mathrm{BE}} = \sum_{n} (d_{\mathrm{est}} \oplus d) + |N_{\mathrm{est}} - N|, \tag{45}$$

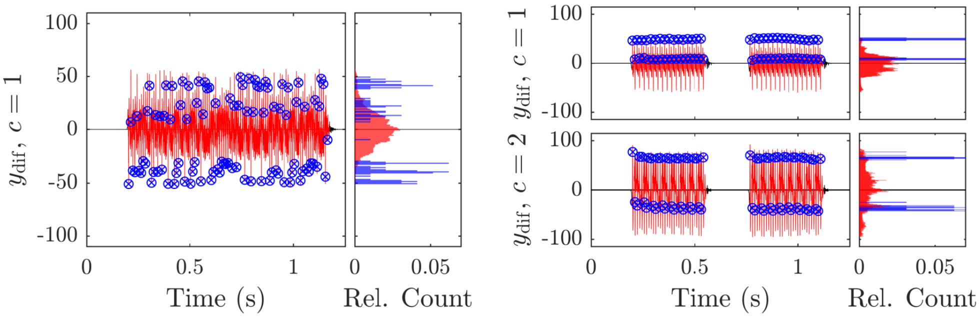
and by convolution, which returns the maximum match between the transmitted sequence d and demodulated sequence d_{est}

$$r_{\mathrm{BExc}} = \max_{n} \{d_{\mathrm{est}} \circledast d\} + |N_{\mathrm{est}} - N|, \tag{46}$$



(a) **(b)**

Figure 9. Averaged spectral power plots of the raw received signals. **Left:** Single band communication; **Right:** Dual-band communication. The colored area marks the $\pm 1\sigma$ region of each frequency bin.



(a) **(b)**

Figure 10. Time domain plot of the detected frames (red) and estimated symbols (blue crossed circles). **Left:** Single band communication; **Right:** Dual-band communication. The symbol difference is not optimally detected, as the amplitude of the signal exceeds the amplitude of the estimated symbols. The histograms to the right of each time plot are normalized by the total number of samples (red) and symbols(blue) in each frame.

both of which include differences in the number of bits to account for additional or missing bits. The former (XOR) we regard for data transmission, where the content of the sequence is not known at the receiver, while the latter (XCorr) indicates the performance, if a known set of codes is expected. The PER is defined through the relative number of erroneous packets compared to the total number of sent packets, where a packet error is any packet that includes at least one bit error. For the PER we consider bit errors according to (45).

The probability of errors approximately follows the error function (*erfc*) over the SNR [52]. While there are closed-form approximations for LoRa [51,53], to ease comparisons we approximate those through superposed error functions

$$\mathrm{P}_{\mathrm{be|pe}}(\gamma) = \sum_q \left(A_q \cdot \mathrm{erfc}\left\{ \left(B_q \cdot 10^{\gamma/10} \right)^{1/2^q} \right\} \right), \tag{47}$$

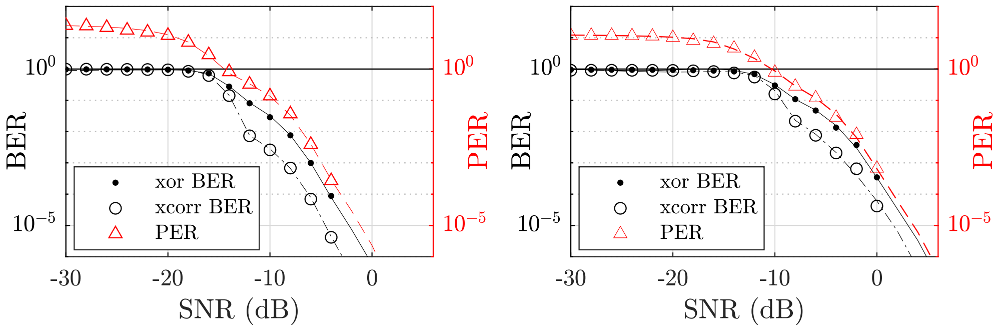
which were fitted manually for the coefficients in Table 3 and Table 4 for the BER and PER simulation, respectively.

Table 3. Single band BER fit coefficients

q		-1	0	1	2
r_{BE}	A_q	0.95	-0.85	0.8	0.5
	B_q	26	50	22	140
r_{BExc}	A_q	1.3	-0.5	0.8	0.02
	B_q	27	50	22	140
r_{PE}	A_q	0	24	2.1	2.0
	B_q	0	50	25	140

Table 4. Dual-band fit coefficients

q	-1	0	1	2
A_q	0.60	-0.65	0.65	0.65
B_q	9.5	50	6.8	43
A_q	1.00	-0.65	0.15	1.00
B_q	10	22	10	65
A_q	4.0	6.0	1.8	1.2
B_q	25	13	7.1	43



(a) **(b)**

Figure 11. Plots of the simulated bit error rate (BER) and packet error rate (PER) for both single band (**left**) and dual-band transmission (**right**).

The dual-band transmission is shifted by approximately 6 dB, which confirms the assumptions from (44). Error rates above 1 indicate that more bits or packets were demodulated than initially sent.

5. Conclusions

The proposed preamble-free Chirp Slope Keying was simulated and tested inside a measurement pool in a laboratory scale experiment for transmission rates of 100 bit/s for single band communications, as well as 200 bit/s for dual-band communication. The bandwidth and symbol length was kept constant. The achieved bit error rate estimated through the bitwise xor operation was simulated to drop below 0.001 for SNR above -6 dB for a time-bandwidth product of 50 in the single band mode and for SNR above 0 dB for a time-bandwidth product of 25 in dual-band mode. The correct detection of packages and the demodulation was successfully implemented, verified and simulated as well. The packet error rate follows the bit error rate with an SNR offset of approximately 1 dB. The simulated channel contained Rayleigh fading and set the SNR through Additive White Gaussian Noise.

Author Contributions: conceptualization, D.J.S. and A.G.; methodology, D.J.S.; software, D.J.S.; validation, D.J.S., A.G., W.X. and G.F.; formal analysis, D.J.S.; investigation, D.J.S.; resources, D.J.S.; data curation, D.J.S.; writing–original draft preparation, D.J.S.; writing–review and editing, D.J.S., A.G., W.X., G.F. and S.J.R.; visualization, D.J.S.; supervision, F.H., J.W., C.S. and S.J.R.; project administration, F.H., J.W., C.S. and S.J.R. funding acquisition, F.H., J.W., C.S. and S.J.R.

Funding: This work has been supported financially under the KMU innovativ initiative of the German Federal Ministry of Education and Research (BMBF), funding number (FKZ) 01IS16010A "ULTa" and 01IS18011C "Smart Diver 4.0", as well as by the state of Baden-Württemberg in the framework of the MERLIN project.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

AF	Active Filter
ALSA	Advanced Linux Sound Architecture
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPF	Bandpass Filter
CSS	Chirp-Spread Sequence
CSP	Chirp Slope Keying
DDC	Digital Down-Converter
DUC	Digital Up-Converter
FDDS	Frame Detect & Downsample
FHX	Fast Hilbert Cross-Correlator
JDS	Join & Downsample
LPF	Lowpass Filter
MUX	Multiplexer
PA	Power Amplifier
PER	Packet Error Rate
RX	Received or Receiver
SNR	Signal-to-Noise Ratio
TB	Time-Bandwidth Product
TX	Transmitted or Transmitter
UAV	Underwater Autonomous Vehicle

References

1. Batchelder, J.M. Submarine Signal. United States 368272, Aug., 1887.
2. Gray, E.; Mundy, Arthur, J. Transmission of Sound. United States 636519, Nov., 1899.
3. Mundy, A.J.; Gale, H.B. Apparatus for Producing Submarine Sound-Signals. United States 842327, Jan., 1907.
4. Stojanovic, M.; Catipovic, J.; Proakis, J.G. Adaptive multichannel combining and equalization for underwater acoustic communications. *The Journal of the Acoustical Society of America* **1993**, *94*, 1621–1631. doi:10.1121/1.408135.
5. Stojanovic, M. Recent advances in high-speed underwater acoustic communications. *IEEE Journal of Oceanic Engineering* **1996**, *21*, 125–136. doi:10.1109/48.486787.
6. Sharif, B.S.; Neasham, J.; Hinton, O.R.; Adams, A.E. A computationally efficient Doppler compensation system for underwater acoustic communications. *IEEE Journal of Oceanic Engineering* **2000**, *25*, 52–61. doi:10.1109/48.820736.
7. Akyildiz, I.F.; Pompili, D.; Melodia, T. Challenges for efficient communication in underwater acoustic sensor networks. *ACM Sigbed Review* **2004**, *1*, 3–8.
8. Steinmetz, F.; Heitmann, J.; Renner, C. A Practical Guide to Chirp Spread Spectrum for Acoustic Underwater Communication in Shallow Waters. Proceedings of the Thirteenth ACM International Conference on Underwater Networks & Systems; Association for Computing Machinery: New York, NY, USA, 2018; WUWNet '18. doi:10.1145/3291940.3291964.
9. Steinmetz, F.; Renner, C. Resilience against Shipping Noise and Interference in Low-Power Acoustic Underwater Communication. OCEANS 2019 MTS/IEEE SEATTLE, 2019, pp. 1–10. doi:10.23919/OCEANS40490.2019.8962790.
10. Zhang, G.; Hovem, J.M.; Dong, H.; Zhou, S.; Du, S. An efficient spread spectrum pulse position modulation scheme for point-to-point underwater acoustic communication. *Applied Acoustics* **2010**, *71*, 11–16.
11. Xu, J.; Li, K.; Min, G. Asymmetric multi-path division communications in underwater acoustic networks with fading channels. *Journal of Computer and System Sciences* **2013**, *79*, 269–278.
12. Xing, S.; Qiao, G.; Tsimenidis, C. A novel two-step Doppler compensation scheme for coded OFDM underwater acoustic communication systems. *Procceedings of the 4th Underwater Acoustics Conference and Exhibition UACE2017* **2017**, pp. 351–356.
13. Domingo, M.C. Overview of channel models for underwater wireless communication networks. *Physical Communication* **2008**, *1*, 163–182. doi:https://doi.org/10.1016/j.phycom.2008.09.001.
14. Hovem, J.M. Underwater acoustics: Propagation, devices and systems. *Journal of Electroceramics* **2007**, *19*, 339–347. doi:10.1007/s10832-007-9059-9.
15. Popper, A.; Hastings, M. The effects of anthropogenic sources of sound on fishes. *Journal of Fish Biology* **2009**, *75*, 455–489. doi:10.1111/j.1095-8649.2009.02319.x.
16. Radford, C.A.; Stanley, J.A.; Tindle, C.T.; Montgomery, J.C.; Jeffs, A.G. Localised coastal habitats have distinct underwater sound signatures. *Marine Ecology Progress Series* **2010**, *401*, 21–29.
17. Schott, D.J.; Faisal, M.; Höeflinger, F.; Reindl, L.M.; Bordoy Andreú, J.; Schindelhauer, C. Underwater localization utilizing a modified acoustic indoor tracking system. 2017 IEEE 7th International Conference on Underwater System Technology: Theory and Applications (USYS), 2017, pp. 1–5. doi:10.1109/USYS.2017.8309451.
18. Khyam, M.O.; Xinde, L.; Ge, S.S.; Pickering, M.R. Multiple Access Chirp-Based Ultrasonic Positioning. *IEEE Transactions on Instrumentation and Measurement* **2017**, *66*, 3126–3137. doi:10.1109/TIM.2017.2737898.
19. Renner, C.; Golkowski, A.J. Acoustic Modem for Micro AUVs: Design and Practical Evaluation. Proceedings of the 11th ACM International Conference on Underwater Networks & Systems; Association for Computing Machinery: New York, NY, USA, 2016; WUWNet '16. doi:10.1145/2999504.3001076.
20. Zhao, Z.; Zhao, A.; Hui, J.; Hou, B.; Sotudeh, R.; Niu, F. A Frequency-Domain Adaptive Matched Filter for Active Sonar Detection. *Sensors* **2017**, *17*. doi:10.3390/s17071565.
21. Bernard, C.; Bouvet, P.J.; Pottier, A.; Forjonel, P. Multiuser Chirp Spread Spectrum Transmission in an Underwater Acoustic Channel Applied to an AUV Fleet. *Sensors* **2020**, *20*. doi:10.3390/s20051527.

22. Danckaers, A.; Seto, M.L. Transmission of images by unmanned underwater vehicles. *Autonomous Robots* **2020**, *44*, 24–44. doi:10.1007/s10514-019-09866-z.
23. Syed, A.A. Understanding and Exploiting the Acoustic Propagation Delay in Underwater Sensor Networks. PhD thesis, Faculty of the Graduate School, CA,USA, 2009.
24. Fattah, S.; Gani, A.; Ahmedy, I.; Idris, M.Y.I.; Targio Hashem, I.A. A Survey on Underwater Wireless Sensor Networks: Requirements, Taxonomy, Recent Advances, and Open Research Challenges. *Sensors* **2020**, *20*. doi:10.3390/s20185393.
25. Hüttmann, E. Distance measurement method. DE 768068C, Jun, 1955.
26. Price, R.; Turin, G. Communication and radar - Section A. *IEEE Transactions on Information Theory* **1963**, *9*, 240–246. doi:10.1109/TIT.1963.1057862.
27. Cook, C.E. Linear FM Signal Formats for Beacon and Communication Systems. *IEEE Transactions on Aerospace and Electronic Systems* **1974**, *AES-10*, 471–478. doi:10.1109/TAES.1974.307800.
28. Burnsweig, J.; Wooldridge, J. Ranging and Data Transmission Using Digital Encoded FM-"Chirp" Surface Acoustic Wave Filters. *IEEE Transactions on Microwave Theory and Techniques* **1973**, *21*, 272–279. doi:10.1109/TMTT.1973.1127978.
29. Kim, J.; Pratt, T.; Ha, T. Coded multiple chirp spread spectrum system and overlay service. The Twentieth Southeastern Symposium on System Theory; IEEE Computer Society: Los Alamitos, CA, USA, 1988; pp. 336–341. doi:10.1109/SSST.1988.17070.
30. TCo. SX1272/3/6/7/8: LoRa Modem Designer's Guide. Technical Report AN1200.13, Semtech Corporation, Camarillo, CA, USA, 2013. Revision 1.
31. Ferré, G.; Giremus, A. LoRa Physical Layer Principle and Performance Analysis. 2018 25th IEEE International Conference on Electronics, Circuits and Systems (ICECS), 2018, pp. 65–68. doi:10.1109/ICECS.2018.8617880.
32. Kebkal, K.G.; Bannasch, R. Sweep-spread carrier for underwater communication over acoustic channels with strong multipath propagation. *The Journal of the Acoustical Society of America* **2002**, *112*, 2043–2052. doi:10.1121/1.1504855.
33. Kaminsky, E.J.; Simanjuntak, L. Chirp slope keying for underwater communications. Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IV; Carapezza, E.M., Ed. International Society for Optics and Photonics, SPIE, 2005, Vol. 5778, pp. 894 – 905. doi:10.1117/12.605426.
34. He, C.; Huang, J.; Zhang, Q.; Lei, K. Reliable Mobile Underwater Wireless Communication Using Wideband Chirp Signal. 2009 WRI International Conference on Communications and Mobile Computing, 2009, Vol. 1, pp. 146–150. doi:10.1109/CMC.2009.25.
35. Demirors, E.; Melodia, T. Chirp-Based LPD/LPI Underwater Acoustic Communications with Code-Time-Frequency Multidimensional Spreading. Proceedings of the 11th ACM International Conference on Underwater Networks & Systems; Association for Computing Machinery: New York, NY, USA, 2016; WUWNet '16. doi:10.1145/2999504.3001083.
36. Yuan, F.; Wei, Q.; Cheng, E. Multiuser chirp modulation for underwater acoustic channel based on VTRM. *International Journal of Naval Architecture and Ocean Engineering* **2017**, *9*, 256–265. doi:https://doi.org/10.1016/j.ijnaoe.2016.09.004.
37. Lee, J.; An, J.; Ra, H.i.; Kim, K. Long-Range Acoustic Communication Using Differential Chirp Spread Spectrum. *Applied Sciences* **2020**, *10*. doi:10.3390/app10248835.
38. Schott, D.J.; Faisal, M.; Höflinger, F.; Reindl, L.M. A Multichannel Acoustic Chirp-Spread Modulation Approach Towards Diver-to-Diver Communication. 2018 15th International Multi-Conference on Systems, Signals Devices (SSD), 2018, pp. 475–479. doi:10.1109/SSD.2018.8570574.
39. Proakis, J.G.; Salehi, M., Elements of a Digital Communication System. chapter 1.1, p. 2. In [54], 2008.
40. Shannon, C.E. A mathematical theory of communication. *The Bell System Technical Journal* **1948**, *27*, 379–423. doi:10.1002/j.1538-7305.1948.tb01338.x.
41. Behar, V.; Adam, D. Parameter optimization of pulse compression in ultrasound imaging systems with coded excitation. *Ultrasonics* **2004**, *42*, 1101–1109. doi:https://doi.org/10.1016/j.ultras.2004.02.020.
42. Proakis, J.G.; Manolakis, D.G., Design of FIR Filters. In *Digital Signal Processing*, 4th ed.; Pearson Prentice Hall: Upper Saddle River, NJ, USA, 2008; chapter 10, p. 668.
43. Cariow, A.; Paplinski, J.P. Some Algorithms for Computing Short-Length Linear Convolution. *Electronics* **2020**, *9*. doi:10.3390/electronics9122115.
44. Aguilera, T.; Álvarez, F.J.; Paredes, J.A.; Moreno, J.A. Doppler compensation algorithm for chirp-based acoustic local positioning systems. *Digital Signal Processing* **2020**, *100*, 102704. doi:https://doi.org/10.1016/j.dsp.2020.102704.
45. Hoeflinger, F.; Hoppe, J.; Zhang, R.; Ens, A.; Reindl, L.; Wendeberg, J.; Schindelhauer, C. Acoustic indoor-localization system for smart phones. 2014 IEEE 11th International Multi-Conference on Systems, Signals Devices (SSD14), 2014, pp. 1–4. doi:10.1109/SSD.2014.6808774.
46. Kastelein, R.A.; Wensveen, P.; Hoek, L.; Terhune, J.M. Underwater hearing sensitivity of harbor seals (Phoca vitulina) for narrow noise bands between 0.2 and 80 kHz. *The Journal of the Acoustical Society of America* **2009**, *126*, 476–483. doi:10.1121/1.3132522.
47. Ketten, D.R. The Marine Mammal Ear: Specializations for Aquatic Audition and Echolocation. The Evolutionary Biology of Hearing; Webster, D.B.; Popper, A.N.; Fay, R.R., Eds.; Springer New York: New York, NY, 1992; pp. 717–750. doi:10.1007/978-1-4612-2784-7_44.
48. Ketten, D.R. Functional analyses of whale ears: adaptations for underwater hearing. Proceedings of OCEANS'94, 1994, Vol. 1, pp. I/264–I/270 vol.1. doi:10.1109/OCEANS.1994.363871.

---

49. Houser, D.S.; Finneran, J.J. A comparison of underwater hearing sensitivity in bottlenose dolphins (Tursiops truncatus) determined by electrophysiological and behavioral methods. *The Journal of the Acoustical Society of America* **2006**, *120*, 1713–1722. doi:10.1121/1.2229286.
50. Stojanovic, M.; Preisig, J. Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Communications Magazine* **2009**, *47*, 84–89. doi:10.1109/MCOM.2009.4752682.
51. Ferreira Dias, C.; Rodrigues de Lima, E.; Fraidenraich, G. Bit Error Rate Closed-Form Expressions for LoRa Systems under Nakagami and Rice Fading Channels. *Sensors* **2019**, *19*. doi:10.3390/s19204412.
52. Proakis, J.G.; Salehi, M., Deterministic and Random Signal Analysis. chapter 2, p. 42. In [54], 2008.
53. Elshabrawy, T.; Robert, J. Closed-Form Approximation of LoRa Modulation BER Performance. *IEEE Communications Letters* **2018**, *22*, 1778–1781. doi:10.1109/LCOMM.2018.2849718.
54. Proakis, J.G.; Salehi, M. *Digital Communications*, 5th ed.; McGraw Hill: New York, NY, USA, 2008.

Sample Availability: Sample recordings of the experiments are available from the authors.