

# Time Reversal Acoustic Communication Receivers: DSP Implementation and Fast Channel Estimation

Sergio Matiz Romero, Mohsen Badiy, and Aijun Song

*smatiz@udel.edu*

---

## Abstract

A communication system is implemented on digital signal processors (DSPs) for the underwater acoustic environment. The implemented receiver uses time reversal multi-channel combining followed by a single-channel decision feedback equalizer. Periodic channel estimation is employed to track the channel fluctuations. These techniques are used to mitigate time-varying inter-symbol interference, which is the main challenge in the underwater acoustic channel at operating frequencies greater than 10 kHz. Various optimization tasks are performed to reduce the receiver computational complexity. A fast implementation of the matching pursuit algorithm is tested on the DSP platform. Its performance, in terms of accuracy and run-time, is compared with that of the basic matching pursuit algorithm. Experimental results of the transmission and demodulation of binary phase-shift keying signals at three different symbol rates were obtained in the local Delaware Bay. The low bit error rates demonstrate the effectiveness of our implementation.

*Keywords:* Time reversal, decision feedback equalizer, channel estimation, underwater acoustic communication.

---

## 1. Introduction

Underwater acoustic (UWA) communication is useful in a variety of industrial and scientific applications such as ocean observation and exploration, sub-sea telemetry, marine security and defense, etc. The UWA channel poses a challenge for high-frequency communication systems due to its severe multipath delay spread and rapid time fluctuations. The substantial inter-symbol interference (ISI) produced by the extensive multipath propagation is difficult to remove, therefore, restricting achievable data rates.

Several methods have been developed to mitigate the time-varying ISI of the underwater acoustic environment. The use of decision feedback equalizer (DFE) for coherent underwater communications has been studied in [1]. Multichannel DFE has been developed in [2] for a multiple access communication network. An alternative method is time reversal [3] or phase conjugation [4, 5]. Time reversal can achieve temporal focusing and compression, thus reducing the ISI. Time reversal followed by equalization has been developed in several

papers [6, 7, 8]. This technique allows for complexity reduction at the equalization stage. Additionally, this type of systems can handle more severe ISI effects than those using only time reversal.

In this paper, we report the implementation of the time reversal DFE receiver from [9] on digital signal processors (DSPs). We utilized the Acoustic Modem Development Platform (AMDP) provided by Aquasent [10] to implement our communication system. To address the time-varying ISI in the oceanic environment, the receiver from [9] organizes the information symbols and demodulates them in a block-by-block fashion. Known symbols are inserted at the beginning of the data packet to perform initial channel estimation and to train the equalizer. We focus on the DSP implementation and develop optimization techniques to reduce the execution time of several processes, such as carrier demodulation and channel estimation. We modify the receiver algorithms in [9] in two ways: use of sparse channel estimators and change of the phase compensation technique. For the former, a fast implementation of the matching pursuit algorithm is utilized for channel estimation to reduce the computational load. For the latter, we remove Doppler/phase compensation at the individual receiving elements for lower complexity. Phase compensation is

---

*Email address:* smatiz@udel.edu (Sergio Matiz Romero, Mohsen Badiy, and Aijun Song)

then conducted after time reversal combining.

A number of efforts have reported acoustic modem implementations. The design of high-data rate UWA systems using field-programmable gate arrays has been described in [11] and [12]. DSP implementation of orthogonal frequency-division multiplexing (OFDM) modems have been reported along with experimental results, for example in [13, 14]. Real-time processing on the DSP has been presented in [13], along with optimization efforts for decoding acceleration. The design of frequency-shift keying UWA communication modems for sensor networks has been described, for instance, in [15, 16]. The development of a configurable modem (phase/frequency-shift keying) for UWA communication has been reported in [17].

Despite widespread availability of acoustic modems, for example commercial products like [18, 10, 19, 20], information on the implementation of UWA modems is still limited. The DSP implementation of time reversal DFE receivers has not been reported in the literature. Thus, one of the contributions of this paper is to provide detailed information on the implementation of this type of receivers. We focus on the optimization techniques that may have a significant impact on the performance of these systems.

This paper is organized as follows. First, an introduction to the UWA channel model is provided in Section 2. The receiver structure and its components are presented. Second, the hardware platform and our DSP implementation are presented in Section 3. The transceiver structure, system operations, and optimization are described in detail. Furthermore, the fast matching pursuit algorithm is presented. Its performance results are compared with those of the basic matching pursuit algorithm. Simulation and experimental results are provided in Section 4.

## 2. System Model and Receiver Structure

The system model and receiver structure are presented in this section. The receiver structure is adopted from [9]. Therefore, the same notations from [9] are used here. Nevertheless, there are some differences in the current receiver. For example, compensation for Doppler and carrier phase offsets is not performed at individual receiving elements for the sake of complexity. Instead, phase compensation is conducted after time reversal. In addition, we use sparse channel estimators, which were not used in [9].

Consider a single-input multiple-output system which consists of one transducer and  $M$  hydrophones. At the acoustic source, a binary symbol sequence  $x(n)$

is modulated onto a carrier frequency  $f_c$  and transmitted. The information signal propagates through the underwater acoustic channel, which introduces ISI, phase fluctuations, and ambient noise to the transmitted signal. On the receiver side, let  $y_m(n)$  be the received baseband signal at the  $m$ th hydrophone. The effect of the UWA medium can be represented by a time-varying channel impulse response (CIR)  $h_m(n, l)$ ,  $0 \leq l \leq L - 1$  and  $1 \leq m \leq M$ , where  $L$  is the discrete channel length. The received signal from a single source at the  $m$ th hydrophone can be written as:

$$y_m(n) = e^{j\theta_m(n)}[x(n) * h_m(n, l)] + v_m(n), \quad (1)$$

where the instantaneous carrier phase offset and ambient noise at the  $m$ th hydrophone are represented by  $\theta_m(n)$  and  $v_m(t)$  respectively. The symbol  $*$  is used to denote convolution. The receiver structure is shown in Fig. 1. It consists of a channel estimator and a demodulator. A detailed description of these two parts and the receiver operation is provided as follows.

### 2.1. Channel Estimator

In Equation (1), the CIR is estimated based on the received signal during an observation period, over which the CIR is assumed constant, that is,  $h_m(n, l) \approx h_m(l)$ . Sparse estimation algorithms are suitable for underwater acoustic channels that have a limited number of significant taps [21, 22]. In this paper, two sparse channel estimation algorithms are considered: basic matching pursuit (BMP), and its modified version, fast matching pursuit (FMP). The FMP is a Fast-Fourier-Transform (FFT) based version of the BMP. The main advantage of the FMP lies in its low complexity. The FMP algorithm is described in detail in Section 3.

### 2.2. Demodulator

Our receiver uses time reversal combining followed by a single-channel DFE to demodulate the transmitted sequence. As shown in Fig. 1, the time reversal block uses  $\hat{h}_m^*(n, -l)$  to filter the incoming signals on each channel  $y_m(n)$  and then the results are combined. The single-channel DFE further reduces the residual ISI in  $z(n)$ . A recursive least squares (RLS) algorithm is used to adaptively update the DFE tap weights. We will refer to the number of DFE feedforward and feedback filter taps as  $N_{ff}$  and  $N_{fb}$  respectively. The RLS forgetting factor is referred to as  $\lambda$ .

The channel estimates are periodically updated to cope with the rapidly time-varying acoustic channel [9, 23, 24]. Fast fluctuating channels require frequent channel updates to approximate the time-varying channel

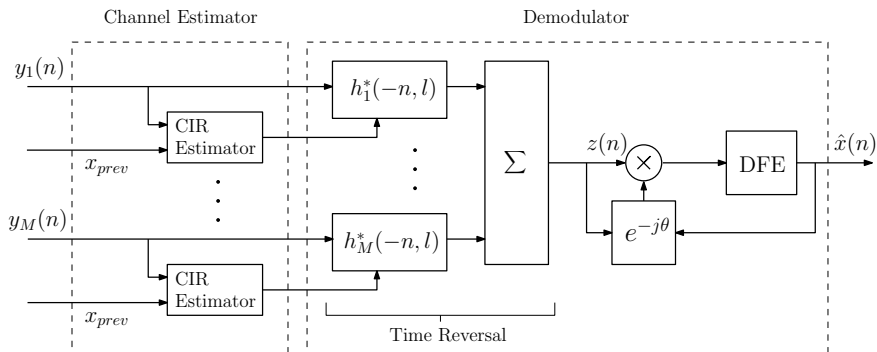


Figure 1: Receiver structure for the time-varying UWA channel.

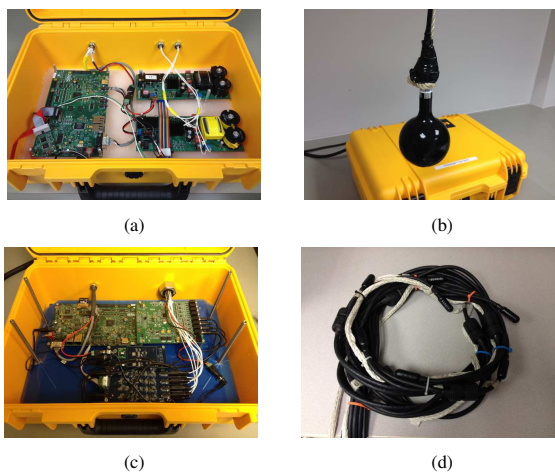


Figure 2: (a) Transmitter box, (b) Transducer, (c) Receiver box, (d) Hydrophone array

impulse response in a piece-wise manner, thereby improving the performance of the receiver. Depending on the characteristics of the channel, the channel update period can range from tens to hundreds of milliseconds [25] [26].

Phase offset compensation is implemented following the method described in [25]. In our system, the phase offset estimates are calculated based on the output of the DFE for the current block. Then the phase correction is applied to the next block.

### 2.3. Receiver Operation

The receiver demodulates the symbol sequence based on channel estimates, which are updated at the channel update period  $T_h$ . The receiver demodulates blocks of  $N_{blk}$  symbols each time the CIR is updated. If the channel update period  $T_h$  is fixed,  $N_{blk}$  can be calculated as  $N_{blk} = T_h R$ , where  $R$  is the symbol rate.

The data packet starts with a preamble of  $N_{preamble}$  known symbols that are used for receiver initialization. This includes tuning of the DFE tap weights via the RLS algorithm and channel estimation. After the training period, the channel estimates are updated based on the demodulation results from the previous block, denoted  $x_{prev}$ . The CIR is considered constant during a period  $T_h$ , when new estimates are calculated. Then the  $N_{blk}$  symbols are processed by the demodulator using the most recent channel estimates. After  $N_{blk}$  symbols of the received sequence are demodulated, the receiver continues with the next block, thereby repeating this process until the whole data packet has been recovered. The BPSK data packet is shown in Fig. 3 (right hand side). The details about signal detection and synchronization are discussed later in Section 3.

Time reversal combining alone can achieve a certain level of equalization effects, as shown in [25]. The combination of time reversal combining and a single channel DFE, as presented in Fig. 1, can achieve enhanced performance. These two configurations are referred to as TR and TR-DFE, respectively.

### 3. DSP Implementation

A pair of AMDPs were used for transmission and reception in our implementation. These are two programmable devices, shown in Fig. 2. The AMDP is controlled by an OMAP-L137 evaluation board, with a Texas Instruments OMAP-L137 as its main processor.

The transmitter and receiver boxes operate on finite state machines (FSMs), which are provided by the vendor [10]. Our implementation was programmed in C language on the OMAP-L137, in each state of the FSMs. In this section, the hardware platform and the FSMs are described. The receiver operations and algorithms are presented. Furthermore, several optimization efforts are detailed.

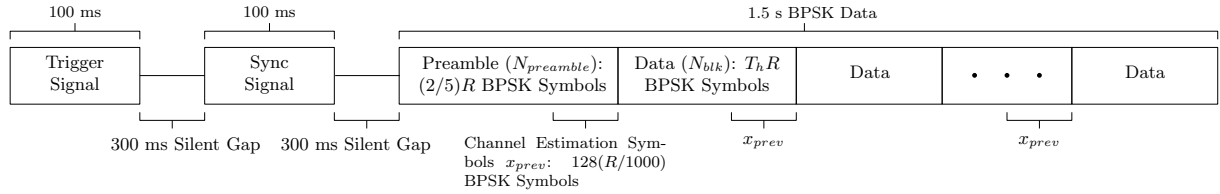


Figure 3: Communication package formation. Symbol rate  $R = 1, 2$  or  $4$  kHz



Figure 4: Transmitter block diagram.

### 3.1. Transmitter Box

The block diagram of the transmitter box is shown in Fig. 4. The DSP evaluation board performs two functions: communication with a host computer and control of the transmission function. It connects with the host computer through the RS232 serial port. In the transmission mode, it receives instructions from the host computer through the serial port. The DSP loads the modulated passband signal, which is sent to the digital-to-analog converter (DAC) to generate the analog signal. Then the analog signal is amplified by the power amplifier to drive the acoustic transducer. The transmitter uses binary phase shift-keying (BPSK) as the modulation scheme. It operates over a 2-state FSM: WAIT and TRANSMIT. In the state WAIT, the transmitter waits for the user input to choose the symbol rate  $R$  for transmission. The symbol rate  $R$  can be 1, 2, or 4 kHz. Notice that, since BPSK modulation is being used, the symbol rate defined in units of Hz is equivalent to the baud rate of the system in units of bits/s. The program remains in the state WAIT until receiving the input for  $R$ . After the input, it goes into the state TRANSMIT. In this state, the transmitter assembles the communication packet waveform at passband, and performs acoustic transmission. The communication packet formation is shown in Fig. 3. The packet consists of a trigger signal, a sync signal, and the BPSK signal. The trigger signal indicates the presence of the communication packet. The sync signal is used for time synchronization. These are both chirp signals, which have a duration of 100 ms and a frequency sweep from 15 to 19 kHz. These three types of signals are separated by silent gaps of 300 ms. The BPSK signal has a duration of 1.5 s and a carrier frequency of 16 kHz. The BPSK symbols are organized in the following manner for a given symbol rate  $R$ . A preamble of  $(2/5)R$  BPSK symbols precedes the

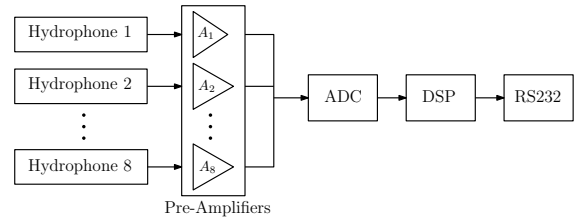


Figure 5: Receiver block diagram.

data packets. Periodic  $128(R/1000)$  BPSK symbols are utilized every  $(1/5)R$ -symbols for channel estimation. After the first block, the channel estimation symbols are obtained from the decoding results of the previous block, thereby forgoing the intermittent training symbols. In the current design, the DSP in the transmitter box loads a pre-generated chirp signal and a BPSK passband waveform. The packet waveform is assembled and sent to the DAC. Then the output of the DAC is amplified to drive the transducer. After the TRANSMIT state, the transmitter goes back to the state WAIT.

### 3.2. Receiver Box

The receiver block diagram is shown in Fig. 5. An array of 8 hydrophones serve as the receiving antennas. The received signal is pre-amplified and fed into the analog-to-digital converter (ADC). The DSP acquires the digitized signals and then proceeds to demodulation. The results of the demodulation can be sent to a computer via the RS232 serial port.

The receiver operations are governed by a five-state FSM: WAIT, GAIN CONTROL, SYNC, DECODE, and CLEAR.

#### 3.2.1. WAIT

During the WAIT state, the receiver waits for the incoming trigger signal. Once it detects the trigger signal, it goes to the state GAIN CONTROL. If no trigger signal is detected, the receiver remains in this state.

#### 3.2.2. GAIN CONTROL

The state GAIN CONTROL is transitional. Based on the strength of the received signal, the gain is adjusted to

modify the signal power to a proper level for the ADC. Then the FSM enters the SYNC state.

### 3.2.3. SYNC

In the SYNC state, the receiver waits for the sync signal. The synchronization process is performed through a matched-filtering operation over the sync signal. Once the synchronization is finished, the receiver goes into the DECODE state.

### 3.2.4. DECODE

In this state, the receiver performs equalization and demodulation of symbols. The DSP obtains the BPSK passband signals from the 8 hydrophones. Carrier demodulation follows to recover the transmitted symbols. The channel estimation algorithms use either the preamble or previously decoded symbols to calculate the CIR estimates. Subsequently, time reversal combining is applied and the first estimate of the transmitted symbols  $z(n)$  is obtained. At the final stage, the DFE is applied to reduce the residual ISI. The RLS algorithm is used to adaptively update the DFE tap values. The demodulation symbol results can be sent to a computer using the RS232 serial port.

### 3.2.5. CLEAR

In this transitional state, the receiver resets itself and goes back to the WAIT state.

## 3.3. Algorithm Optimization

Several optimization efforts were made in our DSP implementation for the receiver algorithms. These efforts aimed to reduce the run-time and to avoid numerical instability.

### 3.3.1. Carrier Demodulation

During carrier demodulation, the signals received by the 8 hydrophones need to be converted to baseband. At each channel, a 1.5 s long BPSK passband signal is sampled at 48 kHz. The carrier demodulation operation includes a mixing operation and a filtering operation. The complexity comes mainly from the filtering operation of passband signals of extended length.

We reduce the complexity of the filtering operation by partitioning the passband signals into smaller blocks. The filtering operation is applied over each one of those partitions. We further reduce the complexity by performing filtering in the frequency domain. The execution time of carrier demodulation on the DSP platform is 6.4 s if no partitioning is used. If 8 partitions are used the execution time is reduced to 3.4 s. This can be further improved to 0.9 s, when using 64 partitions.

Table 1: FMP algorithm description

1: $\mathbf{r}_0 = \mathbf{y}$	▷ Initialize the residual vector
2: $\rho = \ \mathbf{x}\ ^2$	▷ Set the transmitted power
3: $I_0 = \emptyset$	▷ Initialize $I_i$ , the index set of the estimated taps
4: $\mathbf{h}_0 = [0, 0, \dots, 0]^T$	▷ Initial CIR (length $L$ )
5: $\tilde{\mathbf{y}} = \mathcal{F}\{\mathbf{y}\}$	
6: <b>for</b> $i \leftarrow 1, S$ <b>do</b>	▷ $S$ : Sparsity, number of non-zero taps ( $S \ll L$ ).
7: $\tilde{\mathbf{r}}_{i-1} = \mathcal{F}\{\mathbf{r}_{i-1}\}$	
8: $\mathbf{c}_i = \mathcal{F}^{-1}(\tilde{\mathbf{y}} \cdot \tilde{\mathbf{r}}_{i-1})$	▷ Cross-correlation
9: $q_i = \arg \max_{j \notin I_{i-1}} \mathbf{c}_i$	▷ For index $j = \{1, \dots, L\}$
10: $I_i = \{I_{i-1}, q_i\}$	▷ Update $I_i$
11: $\mathbf{h}_i(q_i) = \mathbf{c}(q_i)/\rho$	▷ Obtain $\mathbf{h}_i$
12: $\mathbf{r}_i = \mathbf{y} - \mathcal{F}^{-1}(\tilde{\mathbf{y}} \cdot \tilde{\mathbf{h}}_i)$	▷ Update $\mathbf{r}$
13: <b>end for</b>	

### 3.3.2. Channel Estimation

In time reversal receivers, channel estimation accounts for a significant portion of the receiver complexity since it needs to be performed at each receiving element, repeatedly at the channel update period. In the receiver configuration TR, the channel estimation operations account for more than 90% of the computation. We reduce the complexity of this process by a fast implementation of the sparse channel estimator, which is referred to as the FMP algorithm. Following the scheme in [27], we approximate the matrix-vector multiplication in the BMP through FFT operations. The BMP algorithm aims to find the sparsest solution to the linear system  $\mathbf{y} = \mathbf{C}\mathbf{h} + \boldsymbol{\eta}$ , where  $\mathbf{y} = [y(0), y(1), \dots, y(N-1)]^T$ ,  $\mathbf{h} = [h(0), h(1), \dots, h(L-1)]^T$  and the noise vector  $\boldsymbol{\eta} = [\eta(0), \eta(1), \dots, \eta(N-1)]^T$ . The matrix  $\mathbf{C}$  has dimensions  $N \times L$ . To obtain the solution, the algorithm selects the column of  $\mathbf{C}$  that is best correlated with the residual from the previous iteration, which is denoted as  $\mathbf{r}_{i-1}$  (the subscript  $i$  indicates the value of the vector in the  $i$ -th iteration). This is achieved by solving  $q_i = \arg \max_{j \notin I_{i-1}} |\langle \mathbf{r}_{i-1}, \mathbf{C}_j \rangle|$ , where  $\mathbf{C}_j$  is the  $j$ -th column of matrix  $\mathbf{C}$  and  $I_i$  is the set of selected columns in the  $i$ -th iteration.

Let  $\mathcal{F}$  and  $\mathcal{F}^{-1}$  denote the discrete Fourier transform and its inverse respectively, and let  $\tilde{\mathbf{y}}$  represent the discrete Fourier transform of a vector  $\mathbf{y}$ . The FMP algorithm is based on the assumption that the data Toeplitz matrix  $\mathbf{C}$  can be approximated by a circulant one, which simplifies the BMP optimization problem to  $q_i = \arg \max_{j \notin I_{i-1}} \mathcal{F}^{-1}(\tilde{\mathbf{y}} \cdot \tilde{\mathbf{r}}_{i-1})$ , where the symbol  $\cdot$  denotes component-wise multiplication and  $j$  is the index of the  $j$ -th element of the result of the inverse discrete Fourier transform. For small channel estimation obser-

Table 2: Execution time of the FMP and BMP algorithms for different values of channel length  $L$

$L$	FMP time (ms)	BMP time (ms)	Speed Increase Factor
128	1.6	7.5	4.6
256	3.7	53.6	14.2
512	11.4	431.5	37.7
1024	39.9	3386.2	84.7
2056	153.8	26859.8	174.7

vation windows, this approximation leads to some loss in accuracy.

The complexity of the BMP optimization problem is of the order of  $O(N(L-i-1))$  whereas that of the FMP is of the order of  $O(N \log N)$ . Note that for large values of  $L$ , the FMP algorithm can be far less complex than the BMP one. Moreover, when using the FMP estimation, there is no need to construct and store the matrix  $\mathbf{C}$  on memory.

The detailed FMP algorithm is described in Table 1. The signals  $\mathbf{x}$ ,  $\mathbf{y}$ , and  $\mathbf{h}$  represent the transmitted signal, received signal, and CIR, respectively. The DSP execution times of these two algorithms for different channel lengths are presented in Table 2. The results demonstrate the effectiveness of the FMP algorithm. With larger channel lengths, the execution time reduction becomes more significant. For instance, when  $L = 2056$ , the FMP algorithm is about 174 times faster than the BMP.

To test the accuracy, we compared the BMP and FMP algorithms using a simulated underwater CIR, as shown in Fig. 6. The simulated CIR had extended multipath, with a channel length of  $L = 256$ . The sparse rate  $\alpha$  of the impulse response was set to 0.08 ( $S = \alpha L$ , where  $S$  is the sparsity), i.e., only twenty non-zero coefficients were calculated. In the simulations, the signal-to-noise ratio (SNR) is defined as  $\text{SNR (dB)} = 10 \log_{10}(P_s/\sigma^2)$ , where  $P_s$  is the power of the transmitted signal and  $\sigma^2$  the variance of the noise. The estimated CIRs for an SNR of 15 dB are shown in Fig. 6. Both algorithms attempted to estimate the coefficients that contain the strong channel taps. Some noticeable differences arise around arrival index 180, where the BMP estimate approximates the true CIR more closely than the FMP.

The performance of the two algorithms at different SNRs was also compared, in terms of the mean square error (MSE) between the simulated CIR and its estimates. The SNR varied from 5 to 20 dB. The MSE of the BMP algorithm was smaller than that of the FMP, but difference was relatively small. The MSE of the BMP and the FMP algorithms at an SNR of 15 dB was

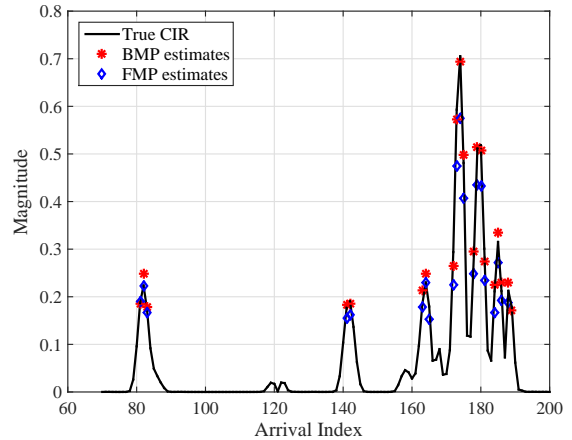


Figure 6: Comparison between the BMP and FMP estimates.

$7.9 \times 10^{-4}$  and  $1.0 \times 10^{-3}$  respectively.

As shown later in Section 4, the performance difference of our communication system using the FMP and BMP algorithms is similar. This makes the FMP a more suitable algorithm for DSP implementations.

### 3.3.3. Numerical Instability

The adaptive DFE requires high numerical accuracy. The RLS algorithm uses past results to update the current filter taps. Errors can propagate and cause unexpected results. We noticed that the division of complex numbers had a significant impact on the performance of the DFE. The RLS algorithm would not converge to an optimal solution and this degraded the receiver performance. To address this issue, the complex division algorithm “Algorithm 116” described in [28] was utilized. Algorithm 116 avoids the squaring of quantities in the denominator, thereby preventing the occurrence of overflow and underflow in the calculations.

## 4. Simulations and Experimental Results

An experiment using the AMDPs was conducted at a shallow water site near the Roosevelt Inlet, Delaware Bay during September, 2014. During the experiment, the transmitter and receiver boxes were deployed 400 m away from each other. The water depth was about 3 m. The hydrophone spacing was 0.25 m. BPSK signals at the symbol rates of 1, 2, and 4 kHz were transmitted. Each symbol rate had three transmissions, referred to as XMT#1, XMT#2, and XMT#3. The transmission and reception parameters are summarized in Table 3.

This section is organized as follows. First, simulations of the receiver performance are presented. We

Table 3: Receiver parameters.

Parameters	Description	Value
$f_c$	Carrier frequency	16 kHz
$R$	Symbol rate	1, 2, or 4 kHz
$L$	Channel length	$128(R/1000)$ symbols
$M$	Number of hydrophones	8
$T_h$	Channel update period	0.2 s
$N_{preamble}$	Length of the preamble	$(2/5)R$ symbols
$N_{blk}$	Demodulation block length	$T_h R$ symbols
$N_{ff}$	DFE number of feedforward filter taps	4
$N_{fb}$	DFE number of feedback filter taps	2
$\lambda$	RLS forgetting factor	0.998
$\alpha$	Channel sparse rate	0.08

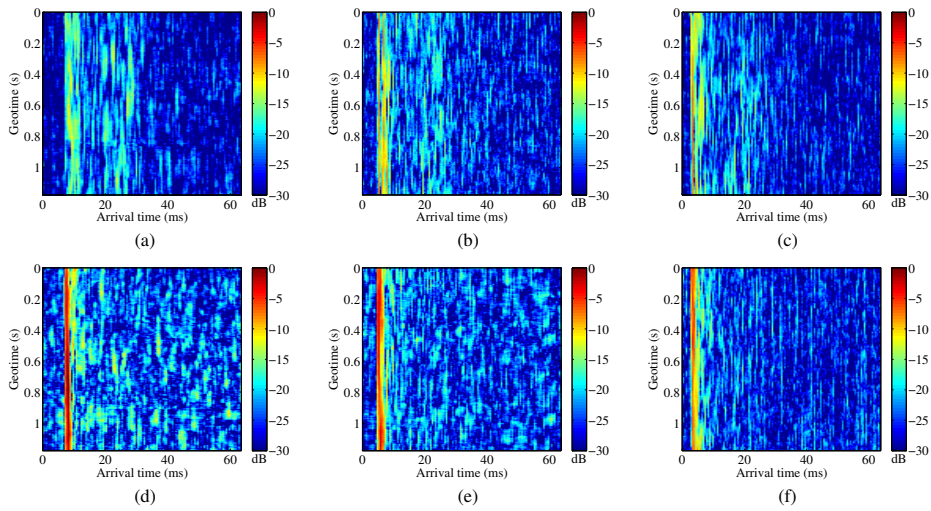


Figure 7: Estimated CIRs for different symbol rates at two hydrophone channels. (a) 1 kHz, (b) 2 kHz, and (c) 4 kHz at channel 2. (d) 1 kHz, (e) 2 kHz, (f) 4 kHz at channel 8.

use the time-varying CIR obtained in the experiment to accurately approximate the underwater channel conditions. The performance is measured in terms of the output MSE and the bit-error-rate (BER) for different values of SNR. Then experimental results of the receiver performance are presented and compared to those of the simulations. To conclude, the execution times of our system are shown.

#### 4.1. Simulations

We simulate the underwater acoustic transmissions using the CIR obtained during our experiment. This allows us to accurately model the performance of our receiver operating over time-varying UWA channels. The simulation program also uses the parameters given in Table 3.

A non-sparse version of the CIR was obtained using the least squares algorithm and it is shown in Fig. 7. The

signal power of channel 2 was lower than that of channel 8. In addition, channel fluctuations can be observed. For example, in Fig. 7(b), the CIR intensity changed during the geotime of 0.2 to 0.6 s. Similarly, in Fig. 7(f), the CIR intensity decreased around the geotime of 0.8 s, compared with that of 0.4 s.

The simulation program first modulates the binary sequence using BPSK. The received symbols are obtained by convolving the BPSK symbols with the CIR and then adding white Gaussian noise with zero mean and variance  $\sigma^2$ . The CIR used by the simulation program was obtained during our experiment using the BMP algorithm, instead of the least squares one, to avoid the noise introduced by the small coefficients in the non-sparse representation of the CIR. The channel estimates were updated every 1 ms to construct the time-varying CIR. The duration of the transmitted signal was set to 1.5 s as in the experiments. At the receiver side, decoding is per-

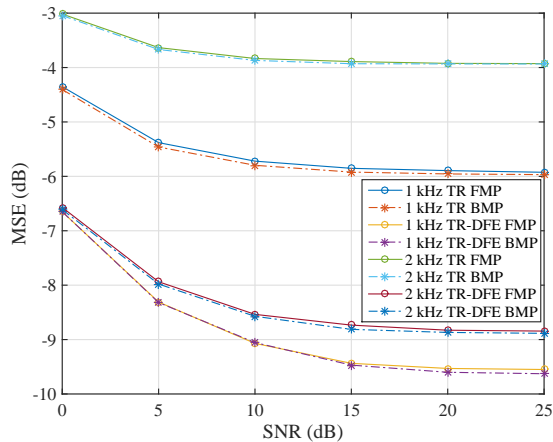


Figure 8: Simulated output MSE as a function of the SNR.

formed using both TR and TR-DFE receivers. We tested these configurations for BMP and FMP channel estimation. The SNR of the transmissions is defined as in Section 3, that is  $\text{SNR (dB)} = 10\log_{10}(P_s/\sigma^2)$ , where  $P_s$  is the power of the transmitted signal and  $\sigma^2$  the variance of the noise. The simulated output MSE of the system as a function of the SNR can be seen in Fig. 8. The curves were obtained by simulating 100 transmissions and then averaging the results. We only show curves for a symbol rate of 1 and 2 kHz for proper visualization, however, similar curves can be obtained for a symbol rate of 4 kHz. Notice that there are two factors that impact the performance of the system, regardless of the type of receiver utilized, namely SNR and symbol rate. The output MSE decreases as the SNR grows and it increases with the symbol rate. The performance improvement of the TR-DFE configuration over the TR one becomes evident in Fig. 8. A difference of about 4 dB between these two configurations can be observed for a symbol rate of 2 kHz when the SNR is 25 dB. It can also be seen that the output MSE is very similar for the BMP and the FMP algorithms. This indicates that the use of the FMP does not lead to performance degradation.

The BER of the system as a function of the SNR is shown in Fig. 9. The BER grows with the symbol rate and decreases with the SNR for both configurations. The performance improvement obtained by using the TR-DFE receiver can also be observed. Notice that the performance of the system, measured in terms of the BER, is also similar for the different channel estimation techniques (BMP and FMP).

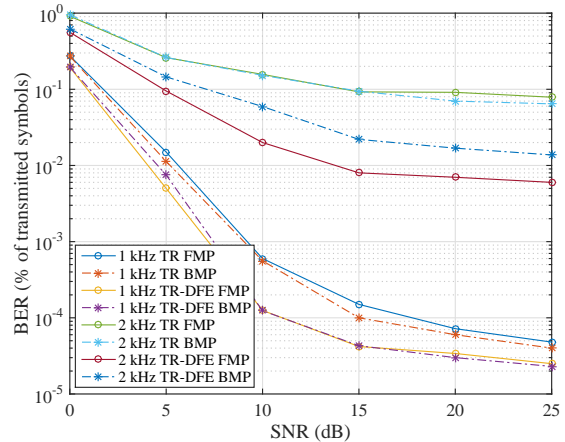


Figure 9: Simulated BER as a function of the SNR.

#### 4.2. Experimental results

The performance results of three transmissions at three different symbol rates obtained during the experiment in the local Delaware bay are presented below. The SNR is calculated following the scheme proposed in [26]. First, the noise energy is obtained using 80 ms of silent gap prior to the BPSK data block. Then the energy of the BPSK data block is calculated for the first 80 ms of data signal. This contains the energy of the signal plus noise. An estimate of the energy of the signal is calculated as the difference between the energy of the BPSK data block and that of the noise. Then the values of signal and noise energy are averaged across the hydrophones. The ratio between the average signal energy over the average noise energy is an estimate of the SNR. During our experiments, the SNR of the transmissions ranged from 23 to 24 dB.

The channel update period  $T_h$  was selected such that the BER of the system in the most challenging conditions would not be significantly degraded, *i.e.*, for a symbol rate of 4 kHz in the TR configuration. Experimental results using FMP channel estimation show that the BER increment from  $T_h = 0.1$  s to  $T_h = 0.2$  s ranges from 10 to 16 %, whereas that of  $T_h = 0.2$  s to  $T_h = 0.4$  s ranges from 68 to 91 %, indicating that for  $T_h > 0.2$  s the performance degradation is severe. The receiver execution time for  $T_h = 0.1$  s is 4.2 s, whereas that of  $T_h = 0.2$  s is 1.9 s. Based on these results, we selected  $T_h = 0.2$  s, since for lower channel update periods the receiver execution time grows with little gain in performance. For  $T_h > 0.2$  s the performance degradation becomes significant. Therefore, the choice of channel update period is an optimization between accuracy and receiver execution time.



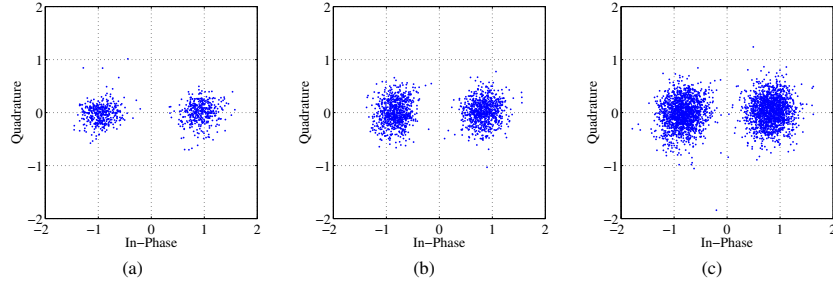


Figure 10: TR-DFE demodulation results for XMT#1 at three symbol rates: (a) 1 kHz, (b) 2 kHz, and (c) 4 kHz

The receiver performance results are summarized in Table 4. Configurations TR and TR-DFE were tested, the results for TR-DFE are shown in parentheses. Both systems used FMP channel estimation. For a symbol rate of 2 kHz, errors occurred only for transmission XMT#3. The maximum BER occurred for the receiver in TR configuration for a symbol rate of 4 kHz during transmission XMT#1, when the BER was 2.13 %. The performance results of the BMP channel estimation are similar to those of the FMP. The biggest difference, in terms of the output MSE, occurred in TR configuration for a symbol rate of 1 kHz during transmission XMT#1. The output MSE for the FMP receiver was -6.15 dB whereas that of the BMP was -6.44 dB. Therefore, as in the case of the simulations, we conclude that the FMP algorithm does not compromise the receiver performance.

Notice that the output MSE was significantly improved when using the TR-DFE configuration. Specially for a symbol rate of 4 kHz, when the difference in output MSE between the TR and TR-DFE configurations was about 6.6 dB. It can also be seen that the results in Table 4 show an upward trend in the output MSE as the symbol rate increases, as was shown previously in the simulation results. Fig. 10 shows the scatter plot of the demodulated symbols at the three different symbol rates when TR-DFE was used. The spread of the demodulated symbols increases as the symbol rate increases, which is consistent with the output MSE.

The experimental results are similar to those obtained in our simulations. We use the simulation results for an SNR of 23 dB for our comparisons, since the SNR in our experiments ranged from 23 to 24 dB. For a symbol rate of 2 kHz, the simulations indicate that the average output MSE of the TR-DFE receiver was -8.8 dB, which is close to the results shown in Table 4, particularly for transmission XMT#2 and XMT#3, where the difference between the simulations and the experimental results is only about 0.2 dB. For a symbol rate of 1 kHz,

Table 4: Performance of the receiver for FMP estimation at three different symbol rates  $R$ . TR and TR-DFE results are shown (TR-DFE results in parentheses).

Transmission number		XMT#1	XMT#2	XMT#3
$R = 1$ kHz	BER (%)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
	MSE (dB)	-6.15 (-10.74)	-6.68 (-9.34)	-5.61 (-9.17)
$R = 2$ kHz	BER (%)	0.00 (0.00)	0.00 (0.00)	0.13 (0.06)
	MSE (dB)	-3.56 (-9.17)	-3.52 (-8.61)	-3.35 (-8.69)
$R = 4$ kHz	BER (%)	2.13 (0.13)	0.44 (0.09)	0.84 (0.03)
	MSE (dB)	-1.42 (-8.06)	-2.99 (-8.49)	-3.12 (-7.36)

Table 5: Demodulation time for the two receiver configurations with the FMP and BMP estimators at three different symbol rates  $R$ .

$R$ (kHz)	Configuration	FMP (s)	BMP (s)
1	TR	1.1	1.6
	TR-DFE	1.8	2.4
2	TR	1.4	4.2
	TR-DFE	3.0	5.7
4	TR	1.9	22.4
	TR-DFE	6.0	25.1

the output MSE for the TR receiver in the simulations was -3.9 dB, which approximates the experimental results of around -3.5 dB. The BER obtained in the experiments for a symbol rate of 1 kHz was always 0.00 %, which agrees with the results shown in Fig. 9. For a symbol rate of 2 kHz, the simulations indicate an average BER of about 0.09 %, for the TR receiver, which is close to the experimental BER of 0.13 % obtained for transmission XMT#3.

The demodulation execution times for the AMDP are shown in Table 5. For both configurations, TR and TR-DFE, the demodulation execution time is significantly reduced when the FMP algorithm is used. The demod-

ulation with the FMP estimator is about 10 times faster than that with the BMP one, when configuration TR is used and the symbol rate is 4 kHz. In configuration TR-DFE, for the same symbol rate, the receiver using the FMP algorithm is about 4 times faster. Notice that for the TR receiver, at symbol rates of 1 and 2 kHz, the demodulation execution time is shorter than the duration of the BPSK data, which is 1.5 s.

## 5. Conclusion

We have implemented high frequency time reversal receivers on a DSP platform for the UWA environment. The architecture and functioning of the communication system over the hardware platform have been presented. In addition, several optimizations have been described. The field experiment demonstrates that these techniques effectively reduce the execution time of the receiver. The FMP algorithm, as a fast implementation of the BMP algorithm, has been introduced and tested on the DSP platform. In comparison with the BMP algorithm, it has been shown that the FMP algorithm is more suitable for DSP implementations, due to a lower complexity and a similar level of accuracy. The low BERs obtained during our experiment at the different symbol rates demonstrate the effectiveness of our DSP implementation.

## 6. Acknowledgment

This research was supported in part by the Office of Naval Research (ONR) Code 322OA through Grants No. N00014-12-1-0789 and N00014-13-1-0306, and the University of Delaware Research Foundation.

## References

- [1] M. Stojanovic, J. Catipovic, J. G. Proakis, Phase-coherent digital communications for underwater acoustic channels, *IEEE J. Ocean. Eng.* 19 (1) (1994) 100–111.
- [2] M. Stojanovic, Z. Zvonar, Multichannel processing of broadband multiuser communication signals in shallow water acoustic channel, *IEEE J. Ocean. Eng.* 21 (2) (1996) 156–166.
- [3] M. Fink, Time-reversed acoustics, *Phys. Today.* 50 (1997) 34–40.
- [4] R. Jackson, D. R. Dowling, Phase conjugation in underwater acoustics, *J. Acoust. Soc. Am.* 89 (1991) 171–181.
- [5] R. Dowling, Acoustic pulse compression using passive phase-conjugate processing, *J. Acoust. Soc. Am.* 95 (1994) 1450–1458.
- [6] G. F. Edelmann, H. C. Song, S. Kim, W. S. Hodgkiss, W. A. Kuperman, T. Akal, Underwater acoustic communications using time reversal, *IEEE J. Oceanic Eng.* 30 (4) (2005) 852–864.
- [7] T. C. Yang, Correlation-based decision-feedback equalizer for underwater acoustic communications, *IEEE J. Oceanic Eng.* 30 (4) (2005) 865–880.
- [8] H. C. Song, W. S. Hodgkiss, W. A. Kuperman, M. Stevenson, T. Akal, Improvement of time reversal communications using adaptive channel equalizers, *IEEE J. Oceanic Eng.* 31 (2) (2006) 487–496.
- [9] A. Song, M. Badiey, H. C. Song, W. S. Hodgkiss, M. B. Porter, K. Group, Impact of ocean variability on coherent underwater acoustic communications during the kauai experiment (kauaiex), *J. Acoust. Soc. Amer.* 123 (2) (2008) 856–865.
- [10] Aquatic Sensor Network Technology. Last accessed in Nov. 2014. URL <http://www.aquasent.com/>
- [11] M. S. Martins, N. Pinto, J. P. Carmo, J. Cabral, High data rate acoustic modem for underwater applications, *International Telecommunications Symposium (ITS), 2014* (2014) 1–5.
- [12] N. Newshean, C. Benson, M. Frater, A high data-rate, software-defined underwater acoustic modem, *OCEANS 2010* (2010) 1–5.
- [13] H. Yan, S. Zhou, Z. Shi, J.-H. Cui, L. Wan, J. Huang, H. Zhou, DSP implementation of SISO and MIMO OFDM acoustic modems, *OCEANS 2010* (2010) 1–6.
- [14] C. L. A. Bourre, S. Lmai, S. Houcke, A robust ofdm modem for underwater acoustic communications, *OCEANS 2013* (2013) 1–5.
- [15] C. R. Berger, Z. Zhou, J. C. Preisig, P. Willett, Sparse channel estimation for multicarrier underwater acoustic communication: from subspace methods to compressed sensing, *IEEE Trans. Sig. Proc.* 58 (3) (2010) 1708–1721.
- [16] A. Sanchez, S. Blanc, P. Yuste, A. Perles, J. Serrano, An ultra-low power and flexible acoustic modem design to develop energy-efficient underwater sensor networks, *Sensors* 2012 12 (2012) 6837–6856.
- [17] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, K. Ball, The whoi micro-modem: an acoustic communications and navigation system for multiple platforms, *OCEANS 2005* (2005) 1086–1092.
- [18] Teledyne Benthos. Last accessed in Jul. 2014. URL <http://www.benthos.com>
- [19] LinkQuest Inc. Last accessed in Oct. 2014. URL <http://www.link-quest.com/>
- [20] EvoLogics. Last accessed in Oct. 2014. URL <http://www.evologics.de/>
- [21] S. F. Cotter, B. D. Rao, Sparse channel estimation via matching pursuit with application to equalization, *IEEE Trans. Commun.* 50 (3) (2002) 374–377.
- [22] W. Li, J. C. Preisig, Estimation of rapidly time-varying sparse channels, *IEEE J. Ocean. Eng.* 32 (4) (2007) 927–939.
- [23] J. Gomes, A. Silva, S. Jesus, Adaptive spatial combining for passive time-reversed communications, *J. Acoust. Soc. Amer.* 124 (2) (2008) 1038–1053.
- [24] A. Song, M. Badiey, A. Newhall, J.F. Lynch, H.A. DeFerrari, B. Katsnelson, Passive time reversal acoustic communications through shallow-water internal waves, *IEEE J. Ocean. Eng.* 35 (4) (2010) 756–764.
- [25] J. A. Flynn, J. A. Ritcey, D. Rouseff, W. L. J. Fox, Multichannel equalization by decision-directed passive phase conjugation : Experimental results, *IEEE J. Ocean. Eng.* 29 (3) (2004) 824–836.
- [26] D. Rouseff, D. R. Jackson, W. L. J. Fox, C. D. Jones, J. A. Ritcey, D. R. Dowling, Underwater acoustic communication by passive-phase conjugation : theory and experimental results, *IEEE J. Oceanic Eng.* 24 (6) (2001) 821–831.
- [27] X. Jiang, W. Zeng, E. Cheng, A fast algorithm for sparse channel estimation via orthogonal matching pursuit, *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd* (2011) 1–5.
- [28] R. L. Smith, *Communications of the ACM* 5 (8).

**Sergio Matiz Romero** received the Bachelor of Electronics Engineering degree from the Pontificia Universidad Javeriana, Bogota, Colombia, in 2010, and the M.S degree in electrical and computer engineering from the University of Delaware, Newark, DE, USA, in 2012. He is currently working toward the PhD degree in electrical and computer engineering at the University of Delaware. His research interests include signal processing, wireless communications and embedded systems design.

**Mohsen Badiey** received the Ph.D. degree in applied marine physics and ocean engineering from University of Miami in Florida, Rosenstiel School of Marine and Atmospheric Science in 1988. He was a Post-Doctoral Fellow at the Port and Harbor Research Institute, Ministry of Transport, Tokyo, Japan, from 1988 through 1990, and worked on research problems related to the water wave interaction with seafloor and on seismic wave propagation on the continental shelf regions. In 1990, he became a faculty member at the College of Earth, Ocean, and Environment, University of Delaware, where he presently is a Professor in the Physical Ocean Science and Engineering Program and in Electrical and Computer Engineering Department. Dr. Badiey's research interests are physics of sound and vibration, underwater acoustics in shallow water regions, acoustical oceanography, underwater acoustic communications, seabed acoustics, and geophysics. He is a Fellow of the Acoustical Society of America.

**Aijun Song** received the Ph.D. degree in electrical engineering from the University of Delaware, Newark, in 2005. From 2005 to 2008, he was a Postdoctoral Research Associate at the College of Earth, Ocean, and Environment (CEOE), University of Delaware. During this period, he was also an Office of Naval Research (ONR) Postdoctoral Fellow supported by the special research award from the Ocean Acoustics program. Since 2008, he has been an Assistant Research Professor of the Physical Ocean Science and Engineering program, University of Delaware. His general interests include underwater acoustic signal propagation, digital communication theory, and advanced signal processing in mobile radio frequency and underwater acoustic environments.

Sergio Matiz Romero



Aijun Song



Mohsen Badiey

