MMP-DCD-CV based Sparse Channel Estimation Algorithm for Underwater Acoustic Transform Domain Communication System

Youwen Zhang\(^a\), Tengfei Wu\(^e\), Yuriy Zakharov\(^d\), Jianghui Li\(^\ast\)

\(^a\)Acoustic Science and Technology Laboratory, Harbin Engineering University, Harbin 150001, China
\(^b\)Key Laboratory of Marine Information Acquisition and Security, Harbin Engineering University, Ministry of Industry and Information Technology, Harbin 150001, China
\(^c\)College of Underwater Acoustic Engineering, Harbin Engineering University, Harbin 150001, China
\(^d\)Department of Electronic Engineering, University of York, York, YO10 5DD, U.K.
\(^e\)Institute of Sound and Vibration Research, University of Southampton, Southampton, SO17 1BJ, U.K.

Abstract
In this paper, we propose a computationally efficient multipath matching pursuit (MMP) channel estimation algorithm for underwater acoustic (UWA) transform domain communication systems (TDCSs). The algorithm, referred to as the MMP-DCD-CV algorithm, is based on the dichotomous coordinate descent (DCD) iterations and cross validation (CV). The MMP-DCD-CV sparse channel estimator in each iteration searches for multiple promising path candidates most relevant to a residual vector and chooses the best candidate. The DCD iterations are used to solve the corresponding least squares problem with low complexity and numerical stability. The CV provides a stopping criterion of the algorithm without a priori information on the channel sparsity and noise level and examines whether the algorithm overfits its data, thus improving the estimation accuracy. The performance of the proposed algorithm is evaluated under simulated sparse UWA channels. The numerical results show that the algorithm achieves better performance than the original MMP algorithm, has lower complexity, and does not require prior knowledge on the channel sparsity and noise level. We also propose an UWA TDCS with sparse channel estimation based on the proposed MMP-DCD-CV algorithm. The proposed UWA communication system is tested by the Waymark simulator, providing the virtual signal transmission in the UWA channel, with a measured Sound Speed Profile and bathymetry. Numerical results demonstrate that the UWA TDCS with the proposed sparse channel estimator offers considerable improvement in system performance compared to other TDCS schemes.

Keywords: Compressive sensing, cross validation, DCD iterations, multipath matching pursuit, sparse channel, transform domain communication system, underwater acoustic communication.

1. Introduction
The underwater acoustic (UWA) channel has a long delay spread, significant Doppler effect, high levels of ambient noise and interference from shipping and marine biological or sonar sources, which pose a great challenge to building reliable and effective underwater communication systems\(^[1, 2, 3, 4]\). Two strategies for interference mitigation (IM) and interference-avoiding (IA) most often used in wireless communications are\(^[5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18]\): modulation format based interference mitigation (MF-IM)\(^[5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]\) and cognitive radio (CR) based interference-avoiding (CR-IA)\(^[16, 17, 18]\).

For the MF-IM schemes, the spread spectrum (SS) modulation is an effective tool to circumvent the interference. Two types of SS techniques have been successfully used in military and civilian wireless communications. Direct sequence based SS (DS-SS) systems use pseudo-random (PR) spreading codes\(^[7, 8,\)]
9, 10, 11], while frequency hopping (FH) based SS (FH-SS) systems use PR hopping sequences to avoid interference[9]. In [14], an interference mitigation receiver is proposed for orthogonal frequency division multiplexing (OFDM) in UWA communications. A generalized likelihood ratio test was used for interference detection, and interference estimation and data decoding were carried out iteratively given the prior knowledge of the frequency band and time duration of the interference. In [15], an iterative receiver for impulsive noise mitigation in UWA OFDM systems is proposed, with positions and amplitude of impulsive noise samples jointly estimated under the least squares (LS) formulation.

Alternatively, the CR-IA schemes[16, 17] perceive the surrounding environment using the CR technology[28] and thus avoid the spectrum interference. Based on this feature, the transform domain communication systems (TDCSs) were proposed[18, 19, 20, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33]. Compared to the SS-based systems, TDCSs improve the robustness against interference by utilizing interference-avoiding waveforms, and information data are modulated on the unoccupied spectrum bins sensed via CR. On the other hand, noise-like basis functions instead of PR spreading codes are used for the information modulation[18], thus, TDCSs possess the same abilities of low interception rate and strong interference avoidance as the SS-based systems[23]. To minimize the computational complexity, the TDCS can be efficiently implemented by the fast Fourier transform (FFT) and inverse FFT (IFFT) operations like that in OFDM systems[27, 28, 29]. The concept of TDCS was first proposed in [18] and then it was applied in a low interception communication system[19]. TDCS was studied for the military communication due to its strong anti-interference ability and a low probability of interception[20, 21, 23, 24, 25, 26]. To the best of our knowledge, there is no research work related to the UWA TDCS.

If we want to apply TDCS underwater, there are two basic problems that should be addressed: 1) interference sensing, and 2) channel state information (CSI) estimation. The interference sensing can be achieved by nonparametric or parametric methods such as the periodogram, multiple signal classification (MUSIC), Capon method, etc.[34]. In this paper, we focus on the CSI estimation, and assume that the narrow band interference is known to both the transmitter and receiver of an UWA TDCS. According to the channel characteristics and adopted modulation format, block-wise or symbol wise adaptive algorithms can be applied for channel estimation. Since the implementation of the TDCS is similar to that of an OFDM system, block-wise channel estimation algorithms are usually utilized. Typical UWA channels have inherent sparsity, i.e., most of the received signal energy arrives at a few channel delays. The traditional LS based channel estimation algorithms are greatly influenced by additive noise and inter carrier interference (ICI). Moreover, there is the noise enhancement problem, so the LS based channel estimation is not appropriate for sparse channels[40, 41, 42, 43, 44].

In recent years, sparse channel estimation based on the compressive sensing (CS) has received a great interest. The CS theory breaks the limitation of the Nyquist sampling theorem. When the signal is sparse in a certain domain[36, 37, 38], one can use fewer sampling points to accurately reconstruct the sparse signal. Due to this advantage, the CS is widely used for the sparse UWA channel estimation[40, 41, 42, 43]. Multiple sparse recovery algorithms have been recently proposed. A sparse signal can be represented by a linear combination of atoms selected from a dictionary[45, 39]. The matching pursuit (MP), at each iteration, selects from a dictionary one atom best matching to a residual signal. An improved selection of atoms is implemented in the orthogonal MP (OMP)[46, 47, 48, 49]. In the MP and OMP algorithms, only one atom is selected at a time. The incorrect atom (a path in the multipath channel estimation) will affect the selection of the next path. The multipath MP (MMP) algorithm searches for multiple possible paths at a time[50]. The candidate set of paths is selected according to the minimum residual energy. The above mentioned sparse recovery algorithms have two common problems: 1) high computational complexity and numerical instability induced by a large number of matrix inverse operations in a sequence of greedy iterations; and 2) requirement for a priori knowledge of the sparsity and noise level.

In this paper, a novel MMP based sparse channel estimation algorithm with dichotomous coordinate descent (DCD) iterations[51] and CV[56, 54, 55, 58, 59, 60, 57], henceforth referred to as MMP-DCD-CV, is proposed for UWA TDCSSs. Our contributions are as follows:

1) We propose a modification to MMP based on the line search with DCD iterations (MMP-DCD).

The solver for an LS problem in the MMP algorithm is based on DCD iterations without explicit
multiplications or divisions, which makes it attractive for real-time implementation on hardware such as FPGAs or DSPs[51, 52]. The complexity of the LS solver in the original MMP algorithm is $O(L^2)$ ($L$ is the size of the LS problem) operations per iteration, whereas the complexity of the DCD based solver is $O(L)$ in the proposed MMP-DCD algorithm.

2) CV is used for estimation of the UWA channel. It is integrated into the MMP-DCD algorithm, leading to the MMP-DCD-CV algorithm. For the original MMP algorithm, the priori information such as sparsity and/or noise level are required to stop greedy iterations. CV is used to estimate the model order of the channel based on the received signal. The reconstruction performance of the proposed MMP-DCD-CV algorithm is evaluated by simulating the virtual signal transmission in sparse UWA channels. Numerical results show that the proposed MMP-DCD-CV algorithm achieves better reconstruction performance than the original MMP algorithm. It also has lower computational complexity and does not require the priori knowledge of the channel sparsity or noise level.

3) We propose an UWA TDCS. The system is built on the proposed MMP-DCD-CV channel estimator to acquire the CSI. To evaluate the performance of proposed UWA TDCS, the receive signal is generated by the Waymark simulator [61, 62, 63, 64] using measured sound speed profile (SSP) and bathymetry. Numerical results show that the proposed UWA TDCS with the MMP-DCD-CV algorithm outperforms other TDCS schemes.

The rest of this paper is organized as follows. In Section II, we present the system model for UWA TDCS. Section III details the proposed sparse channel estimation algorithm for UWA-TDCS. Simulation results based on the Waymark simulator are presented in Section IV. Finally, conclusions are drawn in Section V.

**Notation:** Matrices and vectors are denoted by boldface uppercase and lowercase letters, respectively. $(\cdot)^{T}$ and $(\cdot)^{T}$ denote the Hermitian transposition and the transposition, respectively. $\mathbb{R}$ and $\mathbb{C}$ denote the real field and complex field, respectively. $\circ$, $\odot$ and $\otimes$ denote element-by-element multiplication, the linear convolution and cyclic convolution, respectively. $\|x\|_0$ denotes $l_0$-norm of a vector $x$, $\Re$ denotes the real part of a complex number. $\mathbf{R}(q)$ denotes the $q$-th column of a matrix $\mathbf{R}$. $\mathbf{R}(p,q)$ denotes the element at $p$-th row and $q$-th column of a matrix $\mathbf{R}$. $|S|$ means the cardinality of a set $S$. $(\mathbf{a})_{q}$ denotes a circular shift of elements in the vector $\mathbf{a}$ by $q$ elements.
2. System Model for Underwater Acoustic TDCS

2.1. TDCS Technology

The concept of “transform domain” can be traced back to E. German’s work\[18\] and refers to a transformed version of the received signal in a specified domain (e.g., Fourier, wavelet, etc.) rather than in the domain in which the signal is received\[19, 22, 23\]. For example, the spectral notching technique is a common transform domain filtering technique that removes the narrow band interfering signal by a notch filter\[22, 23\]. However, this interference filtering technique also removes the desired signal energy at notched spectral locations, leading to poor performance of communication compared to the case where no interference is present\[23\]. For TDCSs, both the transmitter and receiver jointly sense contaminated spectrum regions and then avoid using these spectrum regions, i.e., the spectrum locations occupied by the narrow band interference do not carry any information. Therefore the performance of TDCS is not affected by jointly avoiding these contaminated spectrum regions at both the transmitter and receiver\[23, 24, 25\].

2.2. Transmitter with CCSK Modulation

We consider a single-input and single-output UWA TDCS with the cyclic code shift keying (CCSK) modulation. As depicted in Fig. 1, the TDCS modulation procedure involves environment sensing for spectral mask generation, PR phase vector generation, generation of time-domain fundamental modulation waveform (FMW) which is also called the basis function (BF), CCSK modulation, and adding cyclic prefix (CP). The environment sensing unit samples the local acoustic environment and then obtains a spectrum mask vector, $a = [a_0, \cdots, a_n, \cdots, a_{N-1}]$ by using spectrum estimation to sense the unoccupied frequencies. The parameter $N$ is the number of subcarriers or spectrum bins, and $a_n = 1$ means that the $n$-th spectrum bin is unoccupied, otherwise it is occupied\[28\].

A PR phase generator\[35\] is employed to produce a complex PR phase vector $\theta = [\theta_0, \cdots, \theta_i, \cdots, \theta_{N-1}], i = 0, 1, \cdots, N-1$, given by

$$\theta_i = e^{j2\pi m_i/Q}, \quad (2)$$

where $m_i \in \{0, 1, \cdots, Q-1\}$ is a $Q$-ary integer generated by a PR integer generator such as a linear feedback shift register\[35, 21\]. Generally, $Q$ is set to $N$ for the improved performance of CCSK demodulation based on the correlation method. The PR phase vector $\theta$ is multiplied element-by-element with the spectrum mask vector, $a = [a_0, \cdots, a_n, \cdots, a_{N-1}]$ with $a_n \in \{0, 1\}$ by using spectrum estimation to sense the unoccupied frequencies. The parameter $N$ is the number of subcarriers or spectrum bins, and $a_n = 1$ means that the $n$-th spectrum bin is unoccupied, otherwise it is occupied\[28\].

The resultant spectral vector, $b = [b_0, \cdots, b_n, \cdots, b_{N-1}]$ given by

$$b = \lambda a \odot \theta, \quad (3)$$

where $\lambda = \sqrt{E_s/N_T}$ is a magnitude scaling constant to ensure that equal energy symbols are transmitted with the desired signal energy $E_s$, and $N_T = \sum_{n=0}^{N-1} a_n$ is the total number of unoccupied subcarriers. The resultant spectral vector, $b$, is transformed by IFFT into the time domain to produce a noise-like time-domain BF $d = [d_0, \cdots, d_n, \cdots, d_{N-1}]$, $d_n$ given by \[31\]

$$d_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} b_k e^{j\frac{2\pi k n}{N}}, \quad (4)$$

where $b_k = \lambda a_k e^{j\frac{2\pi m_k}{Q}}$.

The data and pilot symbols are multiplexed to produce a payload data vector $c = [c_0, \cdots, c_d, \cdots, c_{D-1}]$ with $c_d \in \{0, 1\}$, where $D$ is the total number of transmission bits in $c$ that consists of data bits and pilots. In SS systems, each transmitted symbol is represented by an unique spreading waveform\[35\]. Whereas, in the binary CCSK modulation, the BF $d$ should be generated from the spectrum mask vector $a$ measured by spectrum sensing before the data transmission and receiving. The spectrum mask will be changed during different communication stages, thus we can not use a pre-specified and fixed BF to modulate and demodulate.
the transmitted and received signal. Therefore, each symbol is represented by the BF $d$ or a circular version of $d$, $(d)_{\tau}$, where $\tau$ is the number to be shifted. In practice, the performance of CCSK demodulation depends on the property of autocorrelation of $d$ and $(d)_{\tau}$, and the property of cross-correlation between $d$ and $(d)_{\tau}$.[26]. The CCSK modulation procedure is seen as a mapping procedure. For example, if $c_d = 0$, the transmitted signal $x = [x_0, x_1, \cdots, x_{N-1}]^T$ is given by[21, 20, 31]

$$x = (d)_0 = [d_0, \cdots, d_n, \cdots, d_{N-1}]^T,$$

(5) otherwise,

$$x = (d)_{\tau=N/2} = [d_0, d_{\tau+1}, \cdots, d_{2\tau-1}, d_0, d_1, \cdots, d_{\tau-1}]^T.$$  

(6)

Details on the $Q$-ary CCSK modulation are referred to [29, 30], and omitted for brevity.

One TDCS symbol is formed by adding CP before $x$ for preventing the interference between successive TDCS symbols, producing the transmit signal $s$.

### 2.3. Receiver with CCSK Demodulation

Here, we assume that both the transmitter and receiver are in the same acoustic interference environment, thus the identical spectrum mask vector $a$ and BF vector $b$ are obtained by environment sensing units at both the transmitter and receiver locations given the same PR phase vector. The CCSK modulated signal is transmitted through the UWA channel. After frame synchronization and removing the CP, the received signal vector $y$ is expressed as

$$y = Hx + n,$$

(7) where $y = [y_0, y_1, \cdots, y_{M-1}]^T$, $M$ is the length of observed samples, $n$ is additive complex white Gaussian noise (AWGN) with zero mean and variance $\sigma_n^2$, which is independent of $x$. The channel convolution matrix $H \in \mathbb{C}^{M \times N}$ is a circular matrix obtained from the channel vector $h = [h_0, h_1, \cdots, h_{L-1}]^T \in \mathbb{C}^{L \times 1}$ by zero padding its $L$ elements to the length $M$ to obtain the first column, where $L$ is the maximum multipath spread in symbol intervals, and $h_l$ denotes the gain of the $l$-th tap in the channel vector[37, 38, 39, 40, 41, 42].

To estimate the UWA channel impulse response $h$, a time-domain pilot block is inserted before transmission of the TDCS symbols. At the receiver, the received signal vector $y_p$ relating to the pilot symbols is extracted out of $y$ by de-multiplexing unit and is given by [37, 38, 39, 40, 41, 42]

$$y_p = p \otimes h + n_p,$$

(8) where $p = [p_0, p_1, \cdots, p_{M-1}]^T$ is the known pilot symbols, $n_p$ is AWGN with zero mean and variance $\sigma_p^2$. Since the cyclic prefix allows converting the linear convolution between the channel impulse response $h$ and transmit signal $p$ into a circular convolution, (8) can be expressed in the cyclic convolution format[37, 38, 39, 40, 41, 42]

$$y_p = p \otimes h + n_p,$$

(9) thus (9) can be rewritten as

$$y_p = C h + n_p,$$

(10) where

$$C = \begin{bmatrix} p_0 & p_1 & \cdots & p_{L-1} \\ p_1 & p_0 & \cdots & p_{L+2} \\ \vdots & \vdots & \ddots & \vdots \\ p_{M-1} & p_{M-2} & \cdots & p_{L+M} \end{bmatrix} \in \mathbb{C}^{M \times L}$$

(11) is a Toeplitz matrix[37, 38, 39, 40, 41, 42].

After channel equalization, the equalized signal is transformed into the time-domain by performing the IFFT operation. Given $b$, $d$, and $\theta$, the estimate of transmitted data can be obtained by performing the CCSK demodulation which is similar to the despreading procedure used in SS systems[35].
3. Proposed MMP-DCD-CV based Sparse Channel Estimation Algorithm

The channel vector in (10) can be estimated by the classical LS method. When the channel vector \( h \) is sparse, the channel estimation problem can be transformed into a sparse reconstruction problem. A number of greedy algorithms were used to estimate the sparse UWA channel over the past decade\cite{40, 41, 42}. On the one hand, given prior information such as sparsity and noise level, most of them can achieve satisfied performance. In practice, such information should be estimated before performing the sparse channel estimation. On the other hand, a large number of matrix inversion operations involved in the LS solver or the greedy algorithms need to be performed. The computational complexity and numerical instability induced by the matrix inversion operations prevent its applications in practical scenarios.

To address aforementioned problems accounted in sparse reconstruction by using greedy algorithms such as OMP etc., motivated by \cite{50, 52, 54, 55, 56, 57, 58, 59, 60}, we propose a new MMP algorithm that combines the DCD iterations and CV. Compared to the OMP algorithm, the MMP algorithm keeps and examines multiple promising candidate support sets rather than retaining only a single path set. Thus the MMP algorithm can overcome the error propagation in the OMP algorithm due to incorrect path selection at a greedy iteration. The DCD iterations can be used to reduce the computational complexity and avoid the numerical instability brought by the matrix inversion operations. The LS solver based on DCD iterations does not require multiplication and division operations\cite{51}, which greatly reduces the computational burden of solving the LS problem, and is particularly suitable for implementation on real-time hardware platforms such as FPGAs or DSPs. The uncertainty on sparsity and noise level can be bypassed using the CV. CV is a statistical method that can check whether the model is correct or not, and avoids underfitting and overfitting of data\cite{54, 55, 56, 58, 59, 60, 57}. Cross validation can be used for stopping greedy iterations without the prior information such as sparsity or noise level.

3.1. Review of the MMP algorithm

For the completeness and understanding of the proposed MMP-DCD-CV algorithm, we review the original MMP algorithm\cite{50}. In the OMP algorithm, at each iteration, only one candidate path that minimizes the residual is added into the support set of the final solution. If there is an incorrect selection of a candidate path during greedy iterations, the resultant solution will be wrong\cite{46, 47, 50}. The MMP algorithm maintains multiple support sets and after finishing all greedy iterations, the best support set of the final solution is chosen from these multiple support sets by minimizing the residuals of the solution\cite{50}.

Table 1 summarizes the original MMP algorithm for sparse channel estimation\cite{50}, where \( K \) is the sparsity level, \( k \) is the iteration index, \( J \) is the number of child paths for each candidate path, \( A^k = \{s_1^k, \cdots, s_i^k, \cdots, s_u^k\} \) is the set of candidate paths at the \( k \)-th iteration, where \( s_i^k \) denotes the \( i \)-th candidate path at the \( k \)-th iteration, \( \hat{h}_u^k \) denotes the estimate of \( h \) at the \( k \)-th iteration for the \( u \)-th candidate path, \( r_i^k \) denotes the residual at the \( k \)-th iteration for the \( u \)-th candidate path, \( C^{s^k} \) and \( C^I \) are the submatrices of \( C \) that contain columns indexed by \( s_i^k \) and \( I \), respectively. The vector \( I \) with \( |I|=J \) denotes the set of all possible combinations of all child paths for all candidate paths at the \( k \)-th iteration, \( \hat{I} = \{\hat{I}_1, \cdots, \hat{I}_J\} \) is the index set of \( L \) columns in \( C \) that are maximally correlated with the residual \( r_i^{k-1} \) at the \( (k-1) \)-th iteration for the \( i \)-th candidate path. For the detailed analysis of the MMP algorithm refer to \cite{50}.

3.2. Proposed MMP-DCD-CV algorithm

According to the theory behind CV, the received vector \( y_p \) in (10) is divided into the reconstruction vector \( y_p^{re} \) and CV vector \( y_p^{cv} \). Correspondingly, the matrix \( C \) is also split into two sub-matrices, a reconstruction matrix \( C_{re} \in \mathbb{C}^{m \times L} \) and a CV matrix \( C_{cv} \in \mathbb{C}^{m_{cv} \times L} \) with \( m_{cv} = M - m \). Then one has a reconstruction equation\cite{55, 56, 57, 58, 59, 60}

\[
y^{re}_p = C_{re} h + n^{re}_p, \tag{12}
\]

and a CV equation

\[
y^{cv}_p = C_{cv} h + n^{cv}_p, \tag{13}
\]
Table 1: MMP Algorithm for Sparse Channel Estimation

**Input:** \( y_p, C, J, K \)

**Initialization:** \( r^0_k = y_p, k = 0, A^0 = \{ \emptyset \} \)

1: while \( k < K \) do
2: \( k = k + 1, u = 0, A^k = \emptyset \)
3: for \( i = 1 \) to \( |A^{k-1}| \) do
4: \( \hat{I} = \arg \max_{|I| = J} \| (C^I)^T r^{k-1}_i \|_2^2 \)
5: for \( j = 1 \) to \( J \) do
6: \( s_{tmp} = s_{k-1}^i \cup \{ \hat{I}_j \} \)
7: if \( s_{tmp} \notin A^k \)
8: \( u = u + 1 \)
9: \( s^k_u = s_{tmp} \)
10: \( A^k = A^k \cup \{ s^k_u \} \)
11: \( h^k_u = (C_{s^k_u})^+ y_p \)
12: \( r^k_u = y_p - C_{s^k_u} h^k_u \)
13: end if
14: end for
15: end for
16: end while
17: \( \hat{u} = \arg \min_{1 \leq u \leq u} \| r^k_u \|_2^2 \)
18: \( \hat{s} = s^K_u \)

**Output:** \( \hat{h} = (C_{\hat{s}})^+ y_p \)

where \( y_p = [y_p^{re T}, y_p^{cv T}]^T \) and \( n_p = [n_p^{re T}, n_p^{cv T}]^T \). The matrix \( C \) is stacked up by \( C_{re} \) and \( C_{cv} \). For a sparse UWA channel, \( h \in C^L \times 1 \) is an unknown \( K \)-sparse channel vector. To exploit the CV in the MMP algorithm, we use \( h^k \) to denote an estimated channel vector in the \( k \)-th iteration, then CV residual of \( h^k \) at the \( k \)-th iteration is given by\([55, 56, 57, 58, 59, 60, 57]\)

\[
\varepsilon^c_{cv k} = \| y_p^{cv} - C_{cv} h^k \|_2^2 .
\]  

(14)

Without CV, i.e., \( m_{cv} = 0 \), we define the residual of \( h^k \) at the \( k \)-th iteration as

\[
\varepsilon_k = \| y_p - Ch_k \|_2^2 .
\]  

(15)

For notational convenience, we may drop the iteration index \( k \) in the following sections.

Table 2 shows a general framework for sparse channel estimation based on the MMP or MMP-DCD algorithm with CV. Instead of setting a maximum number of iterations or threshold for a reconstruction error according to the sparsity or noise level, in algorithms driven by CV, only a maximum number of iterations, \( N_{cv} \), is required. The value of \( N_{cv} \) often can be set two or three times larger than the sparsity \( K \). In practice, this setting value can be coarsely estimated by using the synchronization frame. The sparse channel estimate \( h_k \) is recovered by using the MMP or MMP-DCD algorithm. Given \( h_k \), at the CV equation (14), we can obtain the CV residual of \( h_k \) at the \( k \)-th iteration. After \( N_{cv} \) iterations, the CV residual vector \( \varepsilon \) is used to determine which estimate is the best candidate in the recovered set \( \Omega = \{ h_1, \ldots, h_k, \ldots, h_{N_{cv}} \} \).

The index \( k_{cv} \) corresponding to the minimum CV residual in \( \varepsilon \) is then found, thus indicating the noise- and sparsity-robust channel estimate \( h_{cv} \) in \( \Omega \).

Table 3 details the proposed MMP algorithm that combines the CV and DCD iterations. When using DCD iterations, \( P \) is an upper limit to the amplitude of elements in \( h \). The elements of \( h \) have a fixed-point
representation with $M_b$ bits within the amplitude interval $[-P, P]$; $\delta$ is a step-size; $e$ is the residual vector in DCD iterations. $C_{cs}^k$ and $C_{cs}^k$ denote extracting a column indexed by $s_k^*$ or $I$ from $C_{re}$, respectively. For the choice of parameters related to the MMP algorithm and DCD iterations refer to [50, 51, 52, 59, 60].

4. Numerical results

In this section, the recovery performance of the proposed MMP-DCD-CV algorithm is investigated. The performance of UWA TDCS based on the proposed MMP-DCD-CV algorithm is then evaluated by the Waymark simulator providing the virtual signal transmission in the UWA channel with measured SSP and bathymetry obtained in a real lake experiment as described below.

4.1. Recovery Performance of MMP-DCD-CV algorithm

4.1.1. Exact Reconstruction Probability versus Sparsity Level

To investigate the exact reconstruction probability of the proposed MMP-DCD-CV algorithm, a successful reconstruction condition is defined as follows:

$$\epsilon = \|h - \hat{h}\|^2_2 < \xi_p,$$

where $\epsilon$ is the recovery error of $h$, $\xi_p$ is a small pre-specified constant. The reconstruction is successful if $\epsilon < \xi_p$, otherwise, the reconstruction fails. In this simulation, we consider an underdetermined sparse system, i.e., the dimension $M$ of observation $y$ is smaller than the length $L$ of the input vector $h$. Here, $M$, $L$, $J$, $\sigma^2_p$ and $\xi_p$ are set to 128, 256, 10, 0.1 and $10^{-6}$, respectively. The sparsity level $K$ is considered to be known for OMP and MMP-DCD algorithms over this simulation. The number of bits $M_b$ for representation of the channel taps is set to 32 for both MMP-DCD and MM-DCD-CV algorithms. The exact reconstruction probability of MMP-DCD and MMP-DCD-CV algorithms is shown in Fig. 2. It can be seen that for the same sparsity level $K$ for all algorithms, the MMP-DCD-CV reaches the highest exact reconstruction probability, while that of the OMP algorithm is the lowest. Although the MMP algorithm has the prior information on the sparsity level $K$, its exact reconstruction probability is still inferior to that of the MMP-DCD-CV algorithm. When $K \leq 15$, all the three algorithms achieve the perfect reconstruction. The algorithms completely fail when $K \geq 70$. This simulation demonstrates the high robustness of the MMP-DCD-CV algorithm to recover sparse signals without the prior knowledge on the sparsity.

4.1.2. Exact Reconstruction Probability versus $M_b$ and $K$

Fig. 3 shows how the performance of the MMP-DCD-CV algorithm is affected by the number of bits $M_b$ used in the DCD iterations and the sparsity level $K$ for solving the LS problem. The parameter setup
Table 3: MMP-DCD-CV Algorithm for Sparse Channel Estimation

| Input: $y_p^r$, $y_p^{re}$, $C_{re}$, $C_{cv}$, $J$, $N_{cv}$, $P$, $L$, $M_b$ |
| Initialization: $r_i^0 = y_p^{re}$, $k = 0$, $A^0 = \{\emptyset\}$, $\Omega = \{\emptyset\}$, $s_i^0 = \{\emptyset\}$, $\varepsilon_0^{cv} = \|y_p^{re}\|^2_2$, $0 = C_{re}^i y_p^{re}$, $R = C_{re}^i C_{re}$, $\hat{h}_1 = 0$, $\delta = P$, $\varepsilon = 0_{N_0}$ |
| 1: while $k < N_{cv}$ do |
| 2: $k = k + 1$, $u = 0$, $A^k = \emptyset$ |
| 3: for $i = 1$ to $|A^{k-1}|$ do |
| 4: $I = \arg \max_{I_i \in J} \left\| \left( C_{re}^i \right)^T r_i^{k-1} \right\|^2_2$ |
| 5: for $j = 1$ to $J$ do |
| 6: $s_{tmp} = s_i^k \cup \{ I_j \}$ |
| 7: if $s_{tmp} \notin A^k$ then |
| 8: $u = u + 1$, $s_u^k = s_{tmp}$, $A^k = A^k \cup \{ s_u^k \}$ |
| 9: Using DCD to solve $\hat{h}_u^k = \{ c_{re}^s \}_u^1 y_p^{re}$ |
| 9.1: for $n = 1$ to $M_b$ do |
| 9.2: $\delta = \delta/2$, $\alpha = [\delta, -\delta, j\delta, -j\delta]$ |
| 9.3: flag = 0 |
| 9.4: for $q = 1$ to $L$ do |
| 9.5: for $t = 1$ to $4$ do |
| 9.6: if $\Re \{ \alpha t c_{s}^u \} > R_{(q,t)} \delta^2/2$ then |
| 9.7: $h_q = h_q + \alpha t$ |
| 9.8: $e = e - \alpha t R(q)$ |
| 9.9: flag = 1 |
| 9.10: end if |
| 9.11: end for |
| 9.12: end for |
| 9.13: if flag = 1 then |
| 9.14: go to step 9.4 |
| 9.15: end if |
| 9.16: end for |
| 10: $r_u^k = y_p^{re} - C_{re}^i \hat{h}_u^k$ |
| 11: end if |
| 12: end for |
| 13: end for |
| 14: $\varepsilon(k) = \varepsilon_k^{cv} = \|y_p^{re} - C_{cv} \hat{h}_k\|^2_2$ |
| 15: $\Omega = \Omega \cup \{ \hat{h}_k \}$ |
| 16: end while |
| 17: Compute: $k_{cv} = \arg \min \{ \varepsilon \}$ |

Output: $h = h_{cv}$

is the same as in the previous simulation except for $M_b$ and $K$. We observe that $M_b$ significantly affects the probability of exact reconstruction. It can be seen that, when $M_b$ is less than 22, the algorithm cannot reconstruct the sparse signal vector. With the increase in $M_b$, the probability of exact reconstruction also increases and reaches 100%. This is explained by the fact that $M_b$ defines the quantization error in representing the channel taps. When the error is higher than the variable $\xi_p$ in (16), this inequality cannot be satisfied and the exact reconstruction fails. However, for higher $M_b$, the quantization error is smaller than
\( \xi_p \) in (16) and the reconstruction is successful. For practice, the value \( \xi_p = 10^{-6} \), used in this simulation, is too small, and with higher \( \xi_p \) the number of bits \( M_b \) can be reduced. In practice, we need a trade-off between the probability of exact reconstruction and computational complexity.

\[ \begin{align*}
\text{Exact reconstruction probability} \\
\text{OMP with known } K \\
\text{MMP with known } K \\
\text{MMP-DCD-CV}
\end{align*} \]

\[ \begin{align*}
\text{Exact reconstruction probability} \\
\text{MMP-DCD-CV}, K=5 \\
\text{MMP-DCD-CV}, K=10 \\
\text{MMP-DCD-CV}, K=15 \\
\text{MMP-DCD-CV}, K=20 \\
\text{MMP-DCD-CV}, K=25
\end{align*} \]

4.1.3. Average Recovery Error and Residual versus the Maximum Iteration Number

CV is a statistical technique to identify the model order, avoiding overfitting and underfitting of data. In this simulation, in order to investigate the rationality and effectiveness of the MMP-DCD with CV, \( M, L, K, \sigma_p^2, m_{cv}, \) and \( N_{cv} \) are set to 400, 1000, 50, 0.1, 31, and 150, respectively.

Fig. 4 shows the recovery error \( \epsilon \), residual \( \epsilon \), and CV residual \( \epsilon_{cv} \) as a function of the iteration number \( N_{cv} \) for the MMP-DCD algorithm. It can be seen that the residual \( \epsilon \) decreases monotonically, thus it is...
not obvious when to stop iterations without prior information on the sparsity and noise level. Whereas, the residual of MMP-DCD with CV \( \varepsilon^{cv} \) has a minimum value corresponding to the sparsity level \( N_{cv} \approx K \approx 50 \). At the same time, the residual of MMP-DCD with CV has the same trend as the recovery error. Both the CV residual \( \varepsilon^{cv} \) and recovery error \( \varepsilon \) have the same minimum value point. This simulation further verifies the rationality of CV.

![Figure 4: Residual \( \varepsilon \), CV residual \( \varepsilon^{cv} \), and recovery error \( \varepsilon \) as a function of the maximum iteration number \( N_{cv} \) for the MMP-DCD algorithm.](image)

4.1.4. Average Error versus \( m_{cv} \)

Now we investigate the effect of the dimension of CV equation \( m_{cv} \) on the average error of the MMP-DCD algorithm with the residual and CV residual. In this simulation, the parameters are as follows: \( M, L, K, \sigma^2_p \), and \( N_{cv} \) are set to 400, 1000, 50, 0.1, and 150, respectively. The \( m \) measurements in the matrix \( C \) are used to reconstruct the sparse signal, and the other \( m_{cv} \) measurements are used for CV. The iterations of the MMP-DCD algorithm are stopped by the given sparsity level. As depicted in Fig. 5, even with a small number of CV measurements (about 60), the MMP-DCD-CV algorithm can achieve a performance approaching the performance of the MMP-DCD algorithm with known sparsity and 400 measurements. It can also be seen that the residual of the MMP-DCD algorithm with known sparsity level can not be an indicator to stop the greedy iterations.

4.2. Performance of UWA TDCS with Sparse Channel Estimation based on the MMP-DCD-CV algorithm

4.2.1. Simulation Environment

Fig. 6 depicts the layout of the simulation configuration used in the Waymark simulator. The depth of lake is about 53 m. The transducer and hydrophone were deployed at about 38 m below the surface. The communication distance is 1.5 km. The SSP used in the simulation was measured in the Songhua Lake, Jilin province, China, on November 2013. The bathymetry of the Songhua Lake was obtained by a high resolution multibeam imaging sonar in October 2013. The bottom parameters are: sound speed is 1574 m/s, density is 1.268 g/cm\(^3\), and attenuation coefficient is 0.01875 dB/wavelength. The signaling and data structure is shown in Fig. 7. The carrier frequency is 12 kHz and sampling frequency is 48 kHz. The input bits are encoded by a half rate recursive convolutional code with generator polynomial [171,133] in octal format[65]. Hard decision Viterbi decoder is used[65]. For the CCSK modulation of the UWA TDCS, the indices of \( N_M = N - N_T \) occupied spectrum bins are generated randomly.
4.2.2. Channel Characteristics

To demonstrate how transmission loss (TL) varies with the communication distance, the TL is predicted with the ray tracing software Bellhop [66] using a range-independent waveguide model. Fig. 8 depicts the TL with the measured SSP. The launch angles of the transducer are within the interval $[-30^\circ, 30^\circ]$. It is seen that the TL increases significantly for the ranges of 200 m to 1.5 km. Fig. 9 shows the channel characteristics at 1.5 km. Fig. 9(a) shows the channel impulse response between the source and receiver for
the configuration depicted in Fig. 6. Obviously, the channel is sparse. As shown in Fig. 9(b), there are many
nulls with deep fading over 8 kHz-16 kHz. The multipath spread is long and it is about 90 ms. This type
of channel is very difficult to estimate and equalize.

![Figure 8: Measured SSP and simulated TL.](image)

![Figure 9: UWA channel characteristics: (a) channel impulse response, (b) channel amplitude-frequency response.](image)

4.2.3. Performance of UWA TDCS

Now we investigate the performance of UWA TDCS based on the proposed MMP-DCD-CV algorithm
for varying spread spectrum orders N. Fig. 10 shows the MSE performance of sparse channel estimation
based on the LS, MMP-DCD and MMP-DCD-CV algorithms. The LS and MMP-DCD algorithms operate
assuming the sparsity K to be perfectly known. It can be seen that, with such an assumption, the MMP-
DCD and MMSE-DCD-CV algorithms show similar MSE performance, which is significantly better than
that of the LS algorithm. With increase in N, the MMP-DCD-CV algorithm shows a growing performance
gap between the MMP-DCD and MMP-DCD-CV algorithms with the later outperforming the former. The
gap increases at higher SNR.

Fig. 11 shows the BER performance of UWA TDCS based on the LS, MMP-DCD and MMP-DCD-CV
channel estimators for different spread spectrum orders, with or without coding.
It can be seen that in all cases, the use of the proposed MMP-DCD and MMP-DCD-CV algorithms allows significant improvement of the detection performance provided by the LS algorithm with perfect knowledge of the channel sparsity $K$. The MMP-DCD-CV algorithm, however, shows a better BER performance than the MMP-DCD algorithm, and what is practically very important, it achieves this improvement without a priori knowledge on the channel sparsity and noise level.

As the spread spectrum order $N$ increases, the benefits of using the proposed MMP-DCD-CV algorithm becomes more pronounced in both the MSE and BER performance.

5. Conclusion

In this paper, we have proposed a computationally efficient MMP based sparse channel estimator for an UWA transform domain communication system (TDCS) that should operate without the knowledge of the acoustic channel sparsity and noise level. The estimator is based on the dichotomous coordinate descent (DCD) iterations. The DCD iterations reduce the algorithm complexity and make it numerically stable compared to the LS channel estimator. We have incorporated the CV into the MMP-DCD algorithm and arrived at another algorithm (MMP-DCD-CV), that does not need any prior knowledge on the channel sparsity and noise level. The performance of the proposed MMP-DCD-CV algorithm has been tested using the virtual signal transmission in a sparse UWA channel with a measured SSP and bathymetry. The numerical results verify that better performance is achieved by the proposed MMP-DCD-CV algorithm, with a lower complexity and without a priori knowledge of the channel sparsity and noise level. Numerical
Figure 11: BER performance of the TDCS with different channel estimators and versus the spread spectrum order: (a) \( N = 16 \), (b) \( N = 32 \), and (c) \( N = 64 \).

results show that the proposed UWA TDCS with the proposed sparse channel estimator offers considerable improvement in system performance compared to other TDCS schemes.

Acknowledgment

The work of Y. Zhang and D. Sun were supported in part by the National Natural Science Foundation of China under Grant 61471138 and 61531012, in part by the China Scholarship Council Funding, in part by the Program of International Science and Technology Cooperation under Grant 2013DFR20050, in part by the Defense Industrial Technology Development Program under Grant B2420132004, and in part by the Acoustic Science and Technology Laboratory in 2014. The work of Y. Zakharov is partly supported by the U.K. Engineering and Physical Sciences Research Council under Grant EP/P017975/1 and Grant EP/R003297/1. The work of Jianghui Li was partly supported by the European Union’s Horizon 2020 research and innovation programme under the grant agreement number 654462 (STEMM-CCS).

References


