

Article

Channel Estimation for Underwater Acoustic Communications in Impulsive Noise Environments: A Sparse, Robust, and Efficient Alternating Direction Method of Multipliers-Based Approach

Tian Tian ^{1,2} , Kunde Yang ^{1,2,3,*}, Fei-Yun Wu ⁴  and Ying Zhang ⁵

¹ School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an 710072, China; tiant@mail.nwpu.edu.cn

² Key Laboratory of Ocean Acoustics and Sensing (Northwestern Polytechnical University), Ministry of Industry and Information Technology, Xi'an 710072, China

³ Ocean Institute of Northwestern Polytechnical University, Taicang 215400, China

⁴ Navigation College, Jimei University, Xiamen 361021, China; wfyacademic@gmail.com

⁵ College of Oceanography, Hohai University, Nanjing 210098, China; zying@hhu.edu.cn

* Correspondence: ykdzym@nwpu.edu.cn

Abstract: Channel estimation in Underwater Acoustic Communication (UAC) faces significant challenges due to the non-Gaussian, impulsive noise in ocean environments and the inherent high dimensionality of the estimation task. This paper introduces a robust channel estimation algorithm by solving an $\ell_1 - \ell_1$ optimization problem via the Alternating Direction Method of Multipliers (ADMM), effectively exploiting channel sparsity and addressing impulsive noise outliers. A non-monotone backtracking line search strategy is also developed to improve the convergence behavior. The proposed algorithm is low in complexity and has robust performance. Simulation results show that it exhibits a small performance deterioration of less than 1 dB for Channel Impulse Response (CIR) estimation in impulsive noise environments, nearly matching its performance under Additive White Gaussian Noise (AWGN) conditions. For Delay-Doppler (DD) doubly spread channel estimation, it maintains Bit Error Rate (BER) performance comparable to using ground truth channel information in both AWGN and impulsive noise environments. At-sea experimental validations for channel estimation in Orthogonal Frequency Division Multiplexing (OFDM) systems further underscore the fast convergence speed and high estimation accuracy of the proposed method.

Keywords: robust channel estimation; underwater acoustic communications; compressed sensing; impulsive noise



Citation: Tian, T.; Yang, K.; Wu, F.-Y.; Zhang, Y. Channel Estimation for Underwater Acoustic Communications in Impulsive Noise Environments: A Sparse, Robust, and Efficient Alternating Direction Method of Multipliers-Based Approach. *Remote Sens.* **2024**, *16*, 1380. <https://doi.org/10.3390/rs16081380>

Academic Editors: Fraser Dalgleish and Bing Ouyang

Received: 30 January 2024

Revised: 2 April 2024

Accepted: 8 April 2024

Published: 13 April 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, the expanding scope of marine exploration and the burgeoning Internet of Underwater Things (IoUT) in maritime domains have increased the demand for high-data-rate and reliable wireless communication links [1,2]. Underwater Acoustic (UWA) transmission, leveraging sound waves for their lower attenuation and longer propagation range compared to alternative carriers like electromagnetic and optical waves, is still the most practical solution in ocean environments [3]. However, the development of high-performance Underwater Acoustic Communication (UAC) systems is fundamentally constrained by the intrinsic properties of the UWA channel and the availability of accurate Channel State Information (CSI) [4]. Channel estimation remains a critical challenge in UAC systems. This is compounded by the dynamic and diverse nature of ocean environments and the high dimensionality of the estimation problem due to the large delay and Doppler spreads, along with the need to minimize the number of pilot or training symbols to reduce overhead [5–7].

Extensive research has been conducted on acquiring channel information for various modulation schemes through different methods. Compared to conventional methods such

as Least Squares (LS) [8] and Minimum Mean Squared Error (MMSE) [9], it is widely recognized that Compressed Sensing (CS)-based algorithms offer a more efficient solution for accurate channel estimation with low overhead [10–14]. UWA channels often exhibit sparsity in both delay and Doppler domains when the transmission bandwidth is sufficiently wide to ensure that multipath arrivals are resolvable [10–12]. Iterative greedy algorithms represented by Orthogonal Matching Pursuit (OMP) are the most popular CS-based methods used in practice due to their low computation cost [15,16]. Enhanced variants like Stagewise OMP (StOMP) [17] and the more recent Gram–Schmidt OMP (GSOMP) [13], have improved the efficiency and accuracy of the original OMP by refining atom selection strategies and optimizing termination conditions. However, greedy methods typically yield sub-optimal solutions, and their performance often relies on having prior knowledge of the sparsity level of the channel [18].

Convex relaxation methods, represented by ℓ_1 -norm relaxation or Basis Pursuit Denoising (BPDN), provide more stable solutions in noisy UWA environments, supported by their stronger theoretical guarantees and global optimality [19]. Classic BPDN solvers like the Interior-Point Method (IPM), despite being computationally intensive, are known for their high accuracy and convergence speed [20,21]. For large-scale problems or in scenarios where real-time processing is paramount, advanced solvers such as the Proximal Gradient Method (PGM) [22,23] and the Alternating Direction Method of Multipliers (ADMM) [24,25] offer a desirable compromise between computational efficiency and the precision required for effective channel estimation. Despite the success of the aforementioned methods in UAC applications, their effectiveness is largely predicated on the assumption of Additive White Gaussian Noise (AWGN) [26]. This assumption, however, does not always hold in real-world scenarios as UWA channels often contain interference, especially impulsive noise from various natural sources and human activities [27]. Addressing this, our previous work in [28] introduced a low-complexity, easily implementable ADMM-based algorithm.

Recent developments in Deep Learning (DL) have offered promising solutions to the UWA channel estimation. In [29], a five-layer neural network trained on offline simulated data is developed for channel estimation and symbol detection in wireless Orthogonal Frequency Division Multiplexing (OFDM) systems. For UWA OFDM systems, [30] introduces a DL-based channel estimator that demonstrates improved performance over traditional methods such as LS and MMSE. Meanwhile, DL is also employed in UWA channel prediction. In [31], Liu et al. developed CsiPriNet, combining a 1D Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) network, for frequency-domain channel prediction in UWA downlink OFDMA systems. Despite the advancements in DL approaches, these models often exhibit high latency and demand large storage and computational resources for training.

Achieving robust channel estimation in impulsive noise environments for UAC systems remains an ongoing research endeavor. Conventional channel estimation methods lack resilience against significant outliers, and current approaches to handling impulsive noise often require prior statistical knowledge of the channel and noise, including assumptions about their distribution [32,33]. Additionally, the high computational demands of these methods limit their applicability and effectiveness in real-time applications. To address the challenges outlined above, this work aims to develop a robust and efficient method for UWA channel estimation in impulsive noise environments. Our main contributions are as follows:

1. We introduce a robust algorithm by reformulating the channel estimation problem as an $\ell_1 - \ell_1$ optimization problem, which offers enhanced resilience to outliers. This new optimization problem is adeptly tackled using the ADMM framework with the Accelerated Proximal Gradient (APG) method [34]. Furthermore, we incorporate a non-monotone line search strategy to increase the convergence speed as well as improve the robustness, particularly for ill-conditioned problems. The proposed method is low in complexity and has robust performance in challenging noise conditions.

- We evaluate the performance of the proposed algorithm in various scenarios including estimating Channel Impulse Response (CIR) and Delay-Doppler (DD) spread functions for single-carrier systems through simulations as well as at-sea experimental validations for channel estimation in OFDM systems. The results demonstrate the fast convergence speed and high accuracy of the proposed method in both AWGN and impulsive noise environments, making it well-suited for robust channel estimations for diverse UWA channel models and communication schemes in practical applications.

The remainder of the paper is structured as follows. Section 2 briefly discusses the channel estimation problems in UAC systems. Section 3 details the proposed method, including its implementation and complexity analysis. Numerical simulations and experimental results are in Sections 4 and 5, respectively. Finally, the conclusion is drawn in Section 6.

Notations: Vectors and matrices are represented by boldfaced lower-case and upper-case letters, respectively. The notations $(\cdot)^*$, $(\cdot)^T$, and $(\cdot)^H$ represent complex conjugate, transpose, and Hermitian transpose, respectively. $\langle \cdot, \cdot \rangle$ is an inner product operator, and $\|\cdot\|_p$ represents ℓ_p -norm.

2. Preliminaries

2.1. Channel Estimation in UAC Systems

In UACs, the discrete input–output relationship concerning the transmission of the signal through the channel is often converted to the following linear expression:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x} + \mathbf{v} \quad (1)$$

where $\mathbf{y} \in \mathbb{C}^M$ denotes the received discrete signal or symbols, $\mathbf{x} \in \mathbb{C}^N$ is the vector which encapsulates the unknown channel parameters to estimate, $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ is the matrix formulated by training signal or symbols, and \mathbf{v} corresponds to a noise-related vector. All variables in (1) are presumed to be complex-valued since channel estimations is usually performed on the baseband. Specific definitions of $\mathbf{\Phi}$ and \mathbf{x} depend on the type of the transmitted signal and the channel model adopted. For example, for CIR estimation in single-carrier transmission systems, the discrete input–output relationship of the UWA multipath channel at the i -th time instant can be expressed as:

$$y[i] = \sum_{k=1}^{N_\tau} h^*[k]s[i-k+1] + v[i] \quad (2)$$

where $s[i]$ and $y[i]$ are transmitted and received symbols, respectively, $h[k]$ is the amplitude of the k -th path, and N_τ denotes the number of delay taps or the length of the CIR vector. Let $N = N_\tau$ and $\mathbf{s}[i] = [s[i], s[i-1], \dots, s[i-N+1]]^T$. The matrix form in (1) can be obtained by stacking M consecutive received symbols together, leading to

$$\mathbf{y} \triangleq [y[i], y[i-1], \dots, y[i-M+1]]^T \in \mathbb{C}^M \quad (3a)$$

$$\mathbf{x} \triangleq [h[1], h[2], \dots, h[N]]^T \in \mathbb{C}^N \quad (3b)$$

$$\mathbf{\Phi} \triangleq [\mathbf{s}[i], \mathbf{s}[i-1], \dots, \mathbf{s}[i-M+1]]^T \in \mathbb{C}^{M \times N}. \quad (3c)$$

When the assumption of time-invariant channel does not hold, the input–output relationship in (2) is changed to

$$y[i] = \sum_{k=1}^{N_\tau} h^*[i, k]s[i-k+1] + v[i], \quad (4)$$

where $h[i, k] \triangleq h(i\Delta t, (k-1)\Delta\tau)$ is the sampled time-varying CIR, and Δt and $\Delta\tau$ denote the sampling intervals in time and delay domains, respectively. Instead of using CIR, the time-varying channel can also be represented by its DD spread function $u[l, k] \triangleq u[l, (k-1)\Delta\tau]$

with $\nu_l = \nu_{\min} + (l - 1)\Delta\nu$ for $l = 1, \dots, N_\nu$, and $\Delta\nu$ is the sampling interval in the Doppler domain [35]. The relationship between the DD spread function and the time-varying CIR is given by the Fourier transform:

$$h[i, k] = \sum_{l=1}^{N_\nu} u[l, k] e^{j2\pi\nu_l i \Delta t}. \quad (5)$$

Substituting (5) into (4), we can obtain

$$y[i] = \sum_{k=1}^{N_\tau} \sum_{l=1}^{N_\nu} u^*[l, k] e^{-j2\pi\nu_l i \Delta t} s[i - k + 1] + v[i] \quad (6)$$

which transforms the channel estimation problem to 2-D DD plane. Define the vectorized 2-D matrix of DD spread function $\mathbf{u} \triangleq [u[1, 1], \dots, u[N_\nu, 1], \dots, u[1, N_\tau], \dots, u[N_\nu, N_\tau]]^T$ and the Fourier transform coefficient vector $\boldsymbol{\beta}[i] \triangleq [e^{-j2\pi\nu_1 i \Delta t}, \dots, e^{-j2\pi\nu_{N_\nu} i \Delta t}]^T$. We can rewrite (6) as

$$y[i] = (\mathbf{s}[i] \otimes \boldsymbol{\beta}[i])^T \mathbf{u}^* + v[i], \quad (7)$$

where \otimes represents Kronecker product. Similarly, the time-varying channel estimation via the DD spread function, consistent with the form in (1), can be obtained by stacking M consecutive received symbols with

$$\boldsymbol{\Phi} \triangleq [(\mathbf{s}[i] \otimes \boldsymbol{\beta}[i]), \dots, (\mathbf{s}[i - M + 1] \otimes \boldsymbol{\beta}[i - M + 1])]^T, \quad (8)$$

and $\mathbf{x} \triangleq \mathbf{u} \in \mathbb{C}^N$. Here, $N = N_\tau \times N_\nu$.

Channel estimation problems in multicarrier modulation schemes can also be formulated as the linear expression shown in (1). Different from the single-carrier systems, channel estimation in OFDM systems often operates in the frequency domain, turning time-domain convolutions into straightforward frequency-domain multiplications. Let \mathbf{s} , \mathbf{y} , and \mathbf{w} denote the M -dimensional vectors of transmitted and received pilot symbols and noise in the frequency domain, respectively. Under the assumption that the delay spread of the channel remains within the Cyclic Prefix (CP) length of OFDM symbols, the channel input-output relationship in the frequency domain can be depicted as

$$\mathbf{y} = \boldsymbol{\Lambda} \mathbf{F} \mathbf{h} + \mathbf{w}, \quad (9)$$

where $\boldsymbol{\Lambda} = \text{diag}(\mathbf{s})$ is the diagonal matrix constructed from the transmitted pilot symbols \mathbf{s} , and $\mathbf{F} \in \mathbb{C}^{M \times N}$ is the discrete Fourier transform (DFT) matrix associated with the pilot subcarriers, and $\mathbf{h} = [h[1], h[2], \dots, h[N]]^T$ is the N -dimensional CIR vector. This relationship can be succinctly represented as in (1) by defining $\boldsymbol{\Phi} \triangleq \boldsymbol{\Lambda} \mathbf{F}$ and $\mathbf{x} \triangleq \mathbf{h}$.

In impulsive noise environments, OFDM systems often exhibit better resilience than single-carrier systems, especially when the power of impulsive noise is moderate. This is because symbol detection in the frequency domain averages out impulsive noise across all subcarriers. Conversely, single-carrier systems are more vulnerable to impulsive noise, particularly when the noise occurrence probability is high.

2.2. Compressed Sensing Approach

Above, we briefly discussed how channel estimation problems in UAC, across different channel models and communication schemes, can be converted into a linear programming problem, as shown in (1). A key challenge in solving (1) in the context of UAC is the underdetermined nature of the system. Channel estimation is often performed with fewer training symbols or equations compared to the greater number of unknown channel parameters, i.e., typically $M \ll N$. Fortunately, the inherent sparsity of both CIR and DD spread function of UWA channels makes it possible to solve (1) by using CS-based techniques [6,35]. In Gaussian noise environments, the problem is commonly transformed into the ℓ_1 -norm regularized LS problem

$$\underset{\mathbf{x}}{\text{minimize}} \|\mathbf{y} - \Phi\mathbf{x}\|_2 + \lambda\|\mathbf{x}\|_1, \quad (10)$$

where λ is the regularization parameter. The unconstrained convex optimization problem above can be solved by various optimization algorithms [21,23,25,34,36]. However, as demonstrated in [37], the performance and robustness of sparse estimation algorithms based on (10) will be degraded when the measurement \mathbf{y} is contaminated by impulsive noise or heavy-tailed non-Gaussian noise.

In our preceding work [28], we presented a robust channel estimation method utilizing an alternative objective function

$$\underset{\mathbf{x}}{\text{minimize}} \tau\|\mathbf{y} - \Phi\mathbf{x}\|_1 + \|\mathbf{x}\|_1, \quad (11)$$

which applies ℓ_1 -norm regularization to both solution \mathbf{x} and the residue vector $\mathbf{r} = \mathbf{y} - \Phi\mathbf{x}$, and \mathbf{r} is related to background noise. Compared to (10), optimizing with an $\ell_1 - \ell_1$ objective function provides a more robust solution when the measurement \mathbf{y} contains large errors or impulsive noise, as the squared errors imposed by the ℓ_2 -norm are more sensitive to outliers [38]. $\ell_1 - \ell_1$ optimization is also widely utilized for denoising and restoration in image processing [39,40]. In this work, we introduce an improved ADMM-based method that integrates the APG approach with a non-monotone line search strategy to solve the problem in (11), offering a faster convergence and a more robust and more accurate solution for UWA channel estimation in different scenarios.

3. Details of the Proposed Algorithm

3.1. General Framework

ADMM is an effective method to solve large-scale structured optimization problems in the form

$$\begin{aligned} &\underset{\mathbf{x}, \mathbf{z}}{\text{minimize}} f(\mathbf{x}) + g(\mathbf{z}) \\ &\text{subject to } \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{c} \end{aligned} \quad (12)$$

leveraging the decomposability of dual ascent methods while benefiting from the robust convergence properties of the method of multipliers [25]. Given the objective function in (11), direct optimization is difficult as both ℓ_1 -norm terms are non-differentiable. By introducing an auxiliary variable $\mathbf{z} = \mathbf{y} - \Phi\mathbf{x}$, we can recast (11) into a form amenable to ADMM:

$$\begin{aligned} &\underset{\mathbf{x}, \mathbf{z}}{\text{minimize}} \tau\|\mathbf{z}\|_1 + \|\mathbf{x}\|_1 \\ &\text{subject to } \Phi\mathbf{x} + \mathbf{z} = \mathbf{y}. \end{aligned} \quad (13)$$

The Augmented Lagrangian Function (ALF) of (13) is

$$L_\rho(\mathbf{x}, \mathbf{z}, \boldsymbol{\gamma}) = \|\mathbf{x}\|_1 + \tau\|\mathbf{z}\|_1 + \frac{\rho}{2}\|\mathbf{z} + \Phi\mathbf{x} - \mathbf{y} + \boldsymbol{\gamma}/\rho\|_2^2 - \frac{1}{2\rho}\|\boldsymbol{\gamma}\|_2^2, \quad (14)$$

incorporating the dual variable vector $\boldsymbol{\gamma}$ and a positive penalty factor ρ . Rather than tackling the minimization of $L_\rho(\mathbf{x}, \mathbf{z}, \boldsymbol{\gamma})$ with respect to both \mathbf{x} and \mathbf{z} simultaneously, ADMM decomposes the original problem, allowing for the iterative and alternate updating of \mathbf{x} and \mathbf{z} by

$$\mathbf{x}^{(k+1)} = \underset{\mathbf{x}}{\text{argmin}} L_\rho(\mathbf{x}, \mathbf{z}^{(k)}, \boldsymbol{\gamma}^{(k)}) \quad (15a)$$

$$\mathbf{z}^{(k+1)} = \underset{\mathbf{z}}{\text{argmin}} L_\rho(\mathbf{x}^{(k+1)}, \mathbf{z}, \boldsymbol{\gamma}^{(k)}) \quad (15b)$$

$$\boldsymbol{\gamma}^{(k+1)} = \boldsymbol{\gamma}^{(k)} + \rho(\mathbf{z}^{(k+1)} + \Phi\mathbf{x}^{(k+1)} - \mathbf{y}). \quad (15c)$$

Central to the iteration process above are the two subproblems shown in (15a) and (15b), which, upon transformation, correspond to

$$\underset{\mathbf{x}}{\text{minimize}} F(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x}) \quad (16)$$

$$\underset{\mathbf{z}}{\text{minimize}} G(\mathbf{z}) = g_1(\mathbf{z}) + g_2(\mathbf{z}) \quad (17)$$

with

$$\begin{aligned} f_1(\mathbf{x}) &\triangleq \frac{\rho}{2} \|\mathbf{z}^{(k)} + \Phi \mathbf{x} - \mathbf{y} + \gamma^{(k)}/\rho\|_2^2, & f_2(\mathbf{x}) &\triangleq \|\mathbf{x}\|_1, \\ g_1(\mathbf{z}) &\triangleq \frac{\rho}{2} \|\mathbf{z} + \Phi \mathbf{x}^{(k+1)} - \mathbf{y} + \gamma^{(k)}/\rho\|_2^2, & g_2(\mathbf{z}) &\triangleq \tau \|\mathbf{z}\|_1. \end{aligned} \quad (18)$$

Both $F(\mathbf{x})$ and $G(\mathbf{z})$ comprise a smooth convex quadratic function alongside a non-differentiable ℓ_1 -norm term, which can be solved by PGM. Given a non-differentiable function $h(\mathbf{x})$, its proximal mapping is defined as [36]

$$\text{prox}_{h,t}(\mathbf{x}) = \underset{\mathbf{x}^+}{\text{argmin}} \frac{1}{2t} \|\mathbf{x}^+ - \mathbf{x}\|_2^2 + h(\mathbf{x}^+), \quad (19)$$

where t is a step-size parameter chosen by the Lipschitz constant of the differentiable portion of the objective function. For the ℓ_1 -norm regularization function, its proximal operator is equivalent to the soft-thresholding operator

$$\mathcal{S}_\alpha(\beta) \triangleq \frac{\max(|\beta| - \alpha, 0)}{\max(|\beta| - \alpha, 0) + \alpha} \beta \quad (20)$$

as the separability of the ℓ_1 -norm simplifies the proximal mapping problem into a series of one-dimensional minimization tasks [36]. The following subsections will discuss the detailed schemes for solving the two subproblems.

3.2. Update of Primal Variable: \mathbf{x}

Upon examining (16) and (17) against the standard form of the proximal mapping defined in (19), it is apparent that while the \mathbf{z} -subproblem aligns with the proximal mapping of function $g_2(\mathbf{z})$, the \mathbf{x} -subproblem does not directly correspond due to the existence of matrix Φ . To make the \mathbf{x} -subproblem tractable, we approximate $f_1(\mathbf{x})$ using its first-order Taylor expansion at the point $\mathbf{x}^{(k)}$ —the estimate from the preceding iteration:

$$\tilde{f}_1(\mathbf{x}) \triangleq f_1(\mathbf{x}^{(k)}) + \langle \nabla f_1(\mathbf{x}^{(k)}), \mathbf{x} - \mathbf{x}^{(k)} \rangle + \frac{1}{2t_x} \|\mathbf{x} - \mathbf{x}^{(k)}\|_2^2, \quad (21)$$

where

$$\nabla f_1(\mathbf{x}^{(k)}) = \rho \Phi^H(\mathbf{z}^{(k)} + \Phi \mathbf{x}^{(k)} - \mathbf{y} + \gamma^{(k)}/\rho). \quad (22)$$

Replacing $f_1(\mathbf{x})$ with $\tilde{f}_1(\mathbf{x})$ and omitting the constant terms associated with $\mathbf{x}^{(k)}$, (16) is reduced to an optimization problem equivalent to the proximal mapping of $f_2(\mathbf{x})$ as

$$\mathbf{x}^{(k+1)} = \text{prox}_{f_2, t_x} \left(\mathbf{x}^{(k)} - t_x \nabla f_1(\mathbf{x}^{(k)}) \right), \quad (23)$$

where the step-size parameter t_x can be set as the reciprocal of the Lipschitz constant for $f_1(\mathbf{x})$, i.e., $t_x = 1/\lambda_{\max}(\Phi^H \Phi)$. $\lambda_{\max}(\cdot)$ represents the maximum eigenvalue of the given matrix. The closed-form solution of the proximal operator for ℓ_1 -norm allows (23) to be directly computed by

$$\mathbf{x}^{(k+1)} = \mathcal{S}_{t_x/\rho} \left(\mathbf{x}^{(k)} - t_x \Phi^H(\Phi \mathbf{x}^{(k)} + \mathbf{z}^{(k)} - \mathbf{y} + \gamma^{(k)}/\rho) \right) \quad (24)$$

where $\mathcal{S}_{t_x/\rho}(\cdot)$ is the soft-thresholding function shown in (20) with the threshold parameter $\alpha = t_x/\rho$.

For practical applications, particularly in UWA channel estimation characterized by large delay and Doppler spread, evaluation of $\lambda_{\max}(\Phi^H \Phi)$ can be computationally intensive. In such cases, a backtracking line search strategy is a more flexible and efficient way to adjust the step-size parameters. In [28], we implemented a monotone update scheme for solving the \mathbf{x} -subproblem as detailed in Algorithm 1, where $J(\mathbf{x})$ denotes the objective function given in (11).

Algorithm 1 Update of Primary Variable \mathbf{x} : Monotone Strategy [28]

Iteration 0: Set $\mathbf{x}^{(0)} = \mathbf{0}$, $t_0 > 0$, and $\eta_x > 1$.

...

Iteration k: Get $\mathbf{x}^{(k-1)}$, $\mathbf{z}^{(k-1)}$, $\gamma^{(k-1)}$ and $J(\mathbf{x}^{(k-1)})$ from previous iteration.

1: Compute the gradient $\nabla f_1(\mathbf{x}^{(k-1)})$ using (22).

2: Backtracking Line Search: find the minimum number of iterations $l^{(k)}$ such that with $t_x^{(k)} = t_0/\eta_x^{l^{(k)}}$ and $\mathbf{x}_{l^{(k)}} = S_{t_x^{(k)}/\rho}(\mathbf{x}^{(k-1)} - t_x^{(k)}/\rho \cdot \nabla f_1(\mathbf{x}^{(k-1)}))$

$$J(\mathbf{x}_{l^{(k)}}) \leq J(\mathbf{x}^{(k-1)}).$$

3: Set $\mathbf{x}^{(k)} = \mathbf{x}_{l^{(k)}}$.

In this work, we integrate the APG method to enhance convergence speed [34], characterized by an additional extrapolation step to inject momentum from previous iterations

$$\mathbf{u}^{(k+1)} = \mathbf{x}^{(k)} + \frac{\mu_x^{(k-1)}}{\mu_x^{(k)}}(\mathbf{w}_x^{(k)} - \mathbf{x}^{(k)}) + \frac{\mu_x^{(k-1)} - 1}{\mu_x^{(k)}}(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}), \quad (25)$$

where $\mathbf{w}_x^{(k)}$ is the output of the proximal operator at $\mathbf{u}^{(k)}$ from the previous iteration, and the extrapolation parameter $\mu_x^{(k)}$ is updated by

$$\mu_x^{(k+1)} = \frac{1 + \sqrt{1 + 4(\mu_x^{(k)})^2}}{2}. \quad (26)$$

Accordingly, the proximal gradient step in (23) is adjusted to operate at the new extrapolation point $\mathbf{u}^{(k)}$ as

$$\mathbf{x}^{(k+1)} = \text{prox}_{f_2, t_x}(\mathbf{u}^{(k+1)} - t_x \nabla f_1(\mathbf{u}^{(k+1)})). \quad (27)$$

In ill-conditioned problems or when the optimization landscapes comprise flat plateaus and narrow ravines, enforcing a monotonic decrease in the objective function can slow down the convergence speed [41]. To address this problem, a non-monotone update strategy that allows temporary increases in the objective function value across iterations is introduced. Specifically, we define a relaxation criterion using a convex combination of previous function values $\{J(\mathbf{x}^{(0)}), J(\mathbf{x}^{(1)}), \dots, J(\mathbf{x}^{(k)})\}$:

$$\bar{J}^{(k)} = \frac{\sum_{j=0}^k \varepsilon^{k-j} J(\mathbf{x}^{(j)})}{\sum_{j=0}^k \varepsilon^{k-j}}, \quad (28)$$

where $\varepsilon \in [0, 1)$ is an exponentially decreasing weight controlling the non-monotonicity level. If $\varepsilon = 0$, the strategy reverts to monotone. If $\varepsilon = 1$, $\bar{J}^{(k)}$ is the average function value

from all previous iterations. A similar criterion is also used in non-convex programming to facilitate faster convergence [42]. \bar{J} can also be computed efficiently in a recursive fashion:

$$c^{(k+1)} = \varepsilon c^{(k)} + 1, \quad \bar{J}^{(k+1)} = \frac{\varepsilon c^{(k)} \bar{J}^{(k)} + J(\mathbf{x}^{(k+1)})}{c^{(k+1)}} \quad (29)$$

with $c^{(0)} = 1$ and $\bar{J}^{(0)} = J(\mathbf{x}^{(0)})$. Since we do not require the objective function value to decrease monotonically as in Algorithm 1, the termination condition of the line search process can also be relaxed to reduce the number of inner iteration steps and allow a larger step size to be used. As illuminated in Lemma 2.1 of [23], the linear-quadratic approximation

$$F_s(\hat{\mathbf{x}}, \mathbf{x}^{(k)}) = f_1(\mathbf{x}^{(k)}) + \langle \nabla f_1(\mathbf{x}^{(k)}), \hat{\mathbf{x}} - \mathbf{x}^{(k)} \rangle + \frac{1}{2t_x^{(k+1)}} \|\hat{\mathbf{x}} - \mathbf{x}^{(k)}\|_2^2 + f_2(\hat{\mathbf{x}}) \quad (30)$$

can be used to indicate that a proper step-size $t_x^{(k+1)}$ is found to ensure adherence to the Lipschitz continuity condition.

The proposed non-monotone update strategy of primary variable \mathbf{x} is summarized in the Algorithm 2. To avoid the case that $\mathbf{u}^{(k)}$ might be a bad extrapolation in the APG method, note that we also incorporate an additional normal proximal gradient step to serve as a monitor. This step, as specified in lines 7–9 of Algorithm 2, is activated only when the extrapolation point $\mathbf{u}^{(k)}$ does not contribute to a decrease in the value of the relaxed objective function $\bar{J}^{(k)}$. It ensures the algorithm maintains robust performance and convergence behavior without substantially increasing the computational burden.

Algorithm 2 Update of Primary Variable \mathbf{x} : Non-monotone Strategy

Iteration 0: Set $\mathbf{x}^{(0)} = \mathbf{0}$, $\mu_x^{(0)} = \mu_x^{(-1)} = 1$, $t_0 > 0$, and $\eta_x > 1$.

...

Iteration k: Get $\mathbf{x}^{(k-1)}$, $\mathbf{x}^{(k-2)}$, $\mathbf{z}^{(k-1)}$, $\gamma^{(k-1)}$ and $\bar{J}^{(k-1)}$ from previous iteration.

1: Compute the extrapolation point $\mathbf{u}^{(k)}$ by (25) and the gradient $\nabla f_1(\mathbf{u}^{(k)})$ by (22), respectively.

2: Backtracking Line Search: find the minimum number of iterations $i^{(k)}$ such that with $t_x^{(k)} = t_0/\eta_x^{i^{(k)}}$ and $\mathbf{w}_x^{(k)} = S_{t_x^{(k)}/\rho}(\mathbf{u}^{(k)} - t_x^{(k)}/\rho \cdot \nabla f_1(\mathbf{u}^{(k)}))$

$$F(\mathbf{w}_x^{(k)}) \leq F_s(\mathbf{w}_x^{(k)}).$$

3: **if** $J(\mathbf{w}_x^{(k)}) < \bar{J}^{(k-1)}$ **then**

4: set $\mathbf{x}^{(k)} = \mathbf{w}_x^{(k)}$

5: **else**

6: Compute the gradient $\nabla f_1(\mathbf{x}^{(k-1)})$ by (22).

7: Backtracking Line Search: find the minimum number of iterations $j^{(k)}$ such that with $t_x^{(k)} = t_0/\eta_x^{j^{(k)}}$ and $\tilde{\mathbf{w}}_x^{(k)} = S_{t_x^{(k)}/\rho}(\mathbf{x}^{(k-1)} - t_x^{(k)}/\rho \cdot \nabla f_1(\mathbf{x}^{(k-1)}))$

$$F(\tilde{\mathbf{w}}_x^{(k)}) \leq F_s(\tilde{\mathbf{w}}_x^{(k)}).$$

8: Compute $J(\tilde{\mathbf{w}}_x^{(k)})$ and

$$\mathbf{x}^{(k)} = \begin{cases} \tilde{\mathbf{w}}_x^{(k)}, & \text{if } J(\tilde{\mathbf{w}}_x^{(k)}) < J(\mathbf{w}_x^{(k)}), \\ \mathbf{w}_x^{(k)}, & \text{otherwise.} \end{cases}$$

9: **end if**

10: Update the extrapolation parameter $\mu_x^{(k)}$ by (26), and $c^{(k)}$ and $\bar{J}^{(k)}$ by (29).

3.3. Update of Auxiliary Variable: \mathbf{z}

The optimization of \mathbf{z} in each iteration only includes the evaluation of the proximal operator with constant step-size $t_z = 1$, which is equivalent to performing the element-wise soft-thresholding step as

$$\mathbf{z}^{(k+1)} = S_{\tau/\rho}(\mathbf{z}^{(k)} - \nabla g_1(\mathbf{z}^{(k)})/\rho) \quad (31)$$

with

$$\nabla g_1(\mathbf{z}^{(k)}) = \rho(\mathbf{z}^{(k)} + \Phi \mathbf{x}^{(k+1)} - \mathbf{y} + \gamma^{(k)})/\rho. \quad (32)$$

Similarly, we integrate the extrapolation step of variable \mathbf{z} to improve the convergence rate:

$$\mathbf{v}^{(k+1)} = \mathbf{z}^{(k)} + \frac{\mu_z^{(k-1)}}{\mu_z^{(k)}}(\mathbf{w}_z^{(k)} - \mathbf{z}^{(k)}) + \frac{\mu_z^{(k-1)} - 1}{\mu_z^{(k)}}(\mathbf{z}^{(k)} - \mathbf{z}^{(k-1)}), \quad (33)$$

where $\mathbf{w}_z^{(k)} = S_{\tau/\rho}(\mathbf{v}^{(k)} - \nabla g_1(\mathbf{v}^{(k)})/\rho)$, and $\mu_z^{(k)}$ follows the same update procedure as $\mu_x^{(k)}$ (refer to (26)). The pseudocode of the update process of \mathbf{z} is summarized in Algorithm 3.

Algorithm 3 Update of Auxiliary Variable \mathbf{z}

Iteration 0: Set $\mathbf{z}^{(0)} = \mathbf{y} - \Phi \mathbf{x}^{(0)}$, $\mu_z^{(0)} = \mu_z^{(-1)} = 1$.

...

Iteration k: Get $\mathbf{x}^{(k)}$, $\mathbf{z}^{(k-1)}$, $\mathbf{z}^{(k-2)}$, $\mathbf{w}_z^{(k-1)}$, $\gamma^{(k-1)}$ and $G(\mathbf{z}^{(k-1)})$ from previous iteration.

- 1: Compute the extrapolation point $\mathbf{v}^{(k)}$ by (33) and the gradient $\nabla g_1(\mathbf{v}^{(k)})$ by (32).
 - 2: Evaluate the proximal operator: $\mathbf{w}_z^{(k)} = S_{\tau/\rho}(\mathbf{v}^{(k)} - \nabla g_1(\mathbf{v}^{(k)})/\rho)$.
 - 3: Compute $G(\mathbf{w}_z^{(k)})$ by (17) and $\mathbf{z}^{(k)} = \underset{\mathbf{z}}{\operatorname{argmin}} \{G(\mathbf{z}) : \mathbf{z} = \mathbf{w}_z^{(k)}, \mathbf{z}^{(k-1)}\}$.
 - 4: Update the extrapolation parameter $\mu_z^{(k)}$ by (26).
-

3.4. Residues, Stopping Criteria, and Penalty Parameter Tuning

The convergence of the ADMM algorithm is typically assessed through primal and dual feasibility conditions. Following a similar derivation process of the termination criteria for standard ADMM [25], the primal residue \mathbf{r}_p and dual residue \mathbf{r}_d for our algorithm are given by

$$\mathbf{r}_p^{(k+1)} = \Phi \mathbf{x}^{(k+1)} + \mathbf{z}^{(k+1)} - \mathbf{y}, \quad \mathbf{r}_d^{(k+1)} = \rho \Phi^H (\mathbf{r}_p^{(k+1)} - \mathbf{r}_p^{(k)}) - \frac{1}{t_x^{(k+1)}} (\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}). \quad (34)$$

The iteration stops when the following conditions are met:

$$\|\mathbf{r}_p^{(k)}\|_2 \leq \epsilon_p^{(k)} \quad \text{and} \quad \|\mathbf{r}_d^{(k)}\|_2 \leq \epsilon_d^{(k)} \quad (35)$$

where $\epsilon_p^{(k)}$ and $\epsilon_d^{(k)}$ are updated using absolute and relative criteria:

$$\epsilon_p^{(k)} = \sqrt{M} \epsilon_{\text{abs}} + \epsilon_{\text{rel}} \max\{\|\Phi \mathbf{x}^{(k)}\|_2, \|\mathbf{z}^{(k)}\|_2, \|\mathbf{y}\|_2\}, \quad \epsilon_d^{(k)} = \sqrt{N} \epsilon_{\text{abs}} + \epsilon_{\text{rel}} \|\Phi^H \gamma^{(k)}\|_2. \quad (36)$$

Here, ϵ_{abs} and ϵ_{rel} denote the absolute and relative tolerances, respectively. In addition, to balance the convergence of primal and dual residues and to reduce the sensitivity of the performance to the initial value of penalty parameter ρ , a widely applied adjustment strategy is employed:

$$\rho^{(k+1)} = \begin{cases} \delta^{\text{incr}} \rho^{(k)} & \text{if } \|\mathbf{r}_p^{(k)}\|_2 > \zeta \|\mathbf{r}_d^{(k)}\|_2 \\ \delta^{\text{decr}} \rho^{(k)} & \text{if } \|\mathbf{r}_p^{(k)}\|_2 < \zeta^{-1} \|\mathbf{r}_d^{(k)}\|_2 \\ \rho^{(k)} & \text{otherwise.} \end{cases} \quad (37)$$

Commonly used values for the constants in (37) are $\zeta = 10$ and $\delta^{\text{incr}} = \delta^{\text{decr}} = 2$.

Incorporating the stopping criteria and the penalty parameter adjustment scheme outlined above, the complete pseudocode for the proposed algorithm is summarized in Algorithm 4.

Algorithm 4 Proposed Algorithm

- 1: **Initialization:** Set $\rho, \tau, \varepsilon, \varepsilon_{\text{abs}}, \varepsilon_{\text{rel}} > 0$.
 - 2: **for** $k = 1, 2, 3, \dots$ **do**
 - 3: Update the primary variable $\mathbf{x}^{(k)}$ with the non-monotone strategy in Algorithm 2.
 - 4: Update the auxiliary variable $\mathbf{z}^{(k)}$ by Algorithm 3.
 - 5: Update the dual variable $\gamma^{(k)}$ by (15c).
 - 6: Compute the primal and dual residues by (34) and the tolerances by (36).
 - 7: **if** stopping criteria in (35) are met **then**
 - 8: **break**
 - 9: **end if**
 - 10: Update the penalty parameter $\rho^{(k)}$ by (37).
 - 11: **end for**
-

3.5. Computational Complexity

The proposed algorithm does not include matrix inversion and matrix–matrix multiplication. In the \mathbf{x} -update procedure, the most computationally expensive part of both monotone and non-monotone strategies is the evaluation of the gradient $\nabla f_1(\cdot)$, which takes two matrix–vector multiplications involving Φ and Φ^H , and each multiplication has a complexity of $\mathcal{O}(MN)$. The complexity of the soft-thresholding operator $S_{\lambda_x/\rho}(\cdot)$ in (24) is $\mathcal{O}(N)$. When the extrapolation point \mathbf{u} in (25) does not meet the requirement, the non-monotone strategy will trigger an additional evaluation of gradient vector and soft-thresholding operator. Computation of gradient $\nabla g_1(\cdot)$ only requires performing the matrix–vector product once, which has the same complexity as $\mathcal{O}(MN)$. The evaluation of the soft-thresholding operator in the \mathbf{z} -update has complexity of $\mathcal{O}(M)$. The remaining parts of the proposed algorithm only include vector–vector operations. The overall computational complexity per iteration is $\mathcal{O}(MN)$.

4. Numerical Simulations

In this section, we evaluate the performance of the proposed algorithm for both CIR and DD spread function estimations within the single-carrier transmission scheme. A two-component Gaussian mixture noise (GMN) model [2] is adopted to simulate environments contaminated by impulsive noise with the probability density function denoted as

$$P(\mathbf{v}[i]) = (1 - q)\mathcal{N}(0, \sigma_W^2) + q\mathcal{N}(0, \sigma_I^2), \quad i = 1, \dots, N \quad (38)$$

Here, $\mathbf{v} = \mathbf{v}_W + \mathbf{v}_I$. \mathbf{v}_W and \mathbf{v}_I denote AWGN and impulsive noise with variances σ_W^2 and σ_I^2 , respectively, $0 \leq q < 1$ indicates the impulsive noise occurrence probability, and $\mathcal{N}(\cdot)$ represents the complex Gaussian distribution function. We benchmarked our algorithm against three widely applied first-order methods: OMP, Fast ISTA (FISTA), and the original ADMM developed to solve the ℓ_1 -norm regularized least squares problem in (10). The second-order IPM with Preconditioned Conjugate Gradient (PCG) is also considered [21]. For clarity, the algorithms from our previous work [28] and this study are termed mADMM (monotone) and nmADMM (non-monotone), respectively. All simulations were conducted on a desktop PC manufactured by Dell, located in China, utilizing

an Intel Core i5 2.30 GHz CPU and 16 GB RAM (Intel, Santa Clara, CA, USA), running MATLAB 2023a and Windows 10 (Microsoft, Redmond, WA, USA).

4.1. Simulation I: CIR Estimation in Single-Carrier System

We simulate the UWA channel using the model from [43]. The channel has a total possible delay spread of 128 ms with a sampling interval $\Delta\tau = 0.25$ ms, leading to a CIR of $N = 512$ potential taps. $N_{pa} = 64$ of these taps are non-zero, highlighting the sparsity of the channel. The interarrival times for these non-zero taps are exponentially distributed, and their amplitudes follow a Rayleigh distribution with a variance that decreases as the delay increases. We employ the widely used Pseudo-Random Binary Sequence (PRBS) as the probe signal after binary phase-shift keying (BPSK) modulation. The probe symbol sequence length is set at $M = \frac{1}{2}N = 256$, rendering the problem underdetermined. In the impulsive noise environment, the Interference-to-Noise Ratio (INR) defined as $\text{INR} = 10 \log_{10}(\sigma_I^2/\sigma_W^2)$ is set to 30 dB and the occurrence probability $q = 5 \times 10^{-2}$. To quantify estimation accuracy, we use the Normalized Mean-Square Deviation (NMSD) metric: $\text{NMSD} = 20 \log(\|\mathbf{x}^* - \hat{\mathbf{x}}\|_2/\|\mathbf{x}^*\|_2)$, where \mathbf{x}^* is the true CIR and $\hat{\mathbf{x}}$ denotes its estimate.

Regularization factors for algorithms FISTA, ADMM, and IPM-PCG are all set to $\lambda = 0.01 \lambda_\infty$, where $\lambda_\infty \triangleq \|\mathbf{2}\Phi^H\mathbf{y}\|_\infty$. For the mADMM and nmADMM algorithms, the regularization factor is set as $\tau = 1/(0.05\lambda_\infty)$. Algorithms with a backtracking line search process, specifically FISTA, IPM-PCG, mADMM, and the proposed method, have their scale parameter η for stepsize reduction set to 1.5. The initial penalty parameter ρ for all ADMM-based algorithms is set to 1, and the absolute and relative tolerance thresholds for the stopping criterion are $\epsilon_{\text{abs}} = 10^{-3}$ and $\epsilon_{\text{rel}} = 10^{-2}$, respectively. The additional exponentially decreasing weight parameter which controls the non-monotonicity of the proposed nmADMM algorithm is set as $\epsilon = 0.95$. The stopping criterion for the FISTA algorithm evaluates the relative change ratio of residuals between two consecutive iterations, given as $\|\mathbf{r}^{(k+1)} - \mathbf{r}^{(k)}\|_2/\|\mathbf{r}^{(k)}\|_2$, where $\mathbf{r}^{(k)} = \mathbf{y} - \Phi\mathbf{x}^{(k)}$, and the tolerance for this ratio is $\epsilon_{\text{fista}} = 10^{-3}$. The relative tolerance of the duality gap for the IPM-PCG algorithm is $\epsilon_{\text{ipm}} = 10^{-3}$. Prior to deploying IPM-PCG, the problem is transitioned to the real domain using the method outlined in ref. [21]. All results are averaged over 100 independent runs.

Figures 1 and 2 present an estimated CIR example in two noise environments at a Signal-to-Noise Ratio (SNR) of 10 dB, aligning with the practical conditions in UWA applications. The NMSD versus iteration curves are depicted in Figure 3. One can see that all algorithms achieve a similar NMSD of approximately -12 dB in the AWGN environment. The OMP algorithm can attain the lowest NMSD with precise prior information of sparsity and optimal stopping conditions. The second-order IPM-PCG algorithm offers the fastest convergence; however, its accuracy is compromised due to the problem conversion to the real domain and it has a much higher computational load than other algorithms. In the presence of impulsive noise, the performance of OMP, FISTA, ADMM, and IPM-PCG deteriorates drastically where the estimated CIR contains a large number of noisy taps that are supposed to be inactive. The accuracy of mADMM and nmADMM also decreases by less than 1 dB compared to their performance in the AWGN environment. Yet, they still effectively estimate the channel components. One can also observe that the proposed nmADMM algorithm exhibits a faster convergence speed than the mADMM algorithm, achieving convergence within approximately 20 iterations.

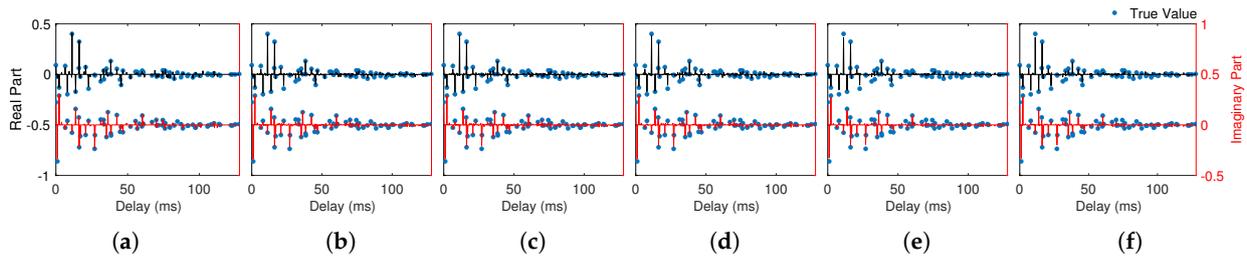


Figure 1. Estimated CIR in AWGN environment: (a) OMP, (b) FISTA, (c) ADMM, (d) IPM-PCG, (e) mADMM, and (f) nmADMM.

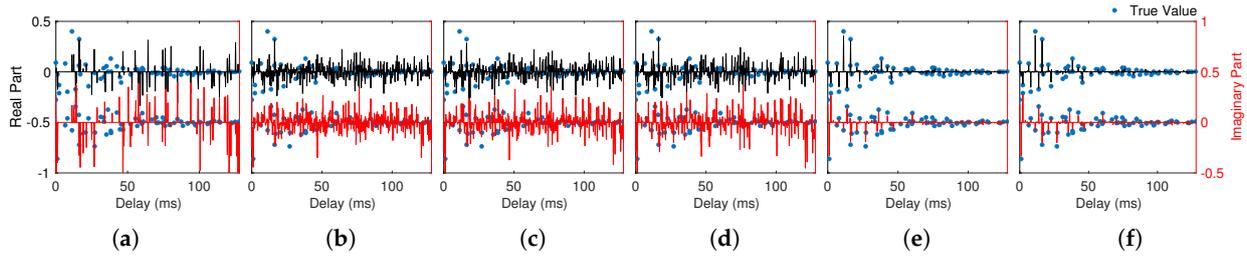


Figure 2. Estimated CIR in impulsive noise environment: (a) OMP, (b) FISTA, (c) ADMM, (d) IPM-PCG, (e) mADMM, (f) nmADMM.

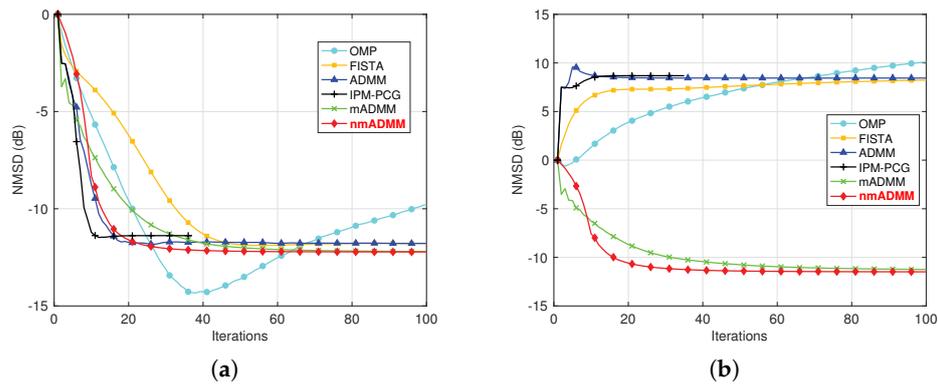


Figure 3. Comparison of convergence trends of NMSD across iterations: (a) AWGN environment and (b) impulsive noise environment.

Figure 4 shows the variation in NMSD with a different number of measurements, which further highlights the robust performance of the proposed method in impulsive noise environments. Table 1 summarizes the average number of iterations to reach stop criteria, runtime for single estimation, and converged NMSD values. While the proposed non-monotone strategy requires longer computation due to the extra proximal step, it excels in convergence speed and estimation accuracy in both AWGN and impulsive noise environments compared to the mADMM algorithm. Figure 5 displays the NMSD across various SNR levels. In AWGN environments, with precise channel sparsity information, the OMP algorithm obtains the lowest NMSD at high SNR levels (≥ 10 dB), while other algorithms perform similarly. However, in impulsive noise environments, both the mADMM and nmADMM algorithms outperform other competing algorithms by over 15 dB across all SNR levels. Notably, the proposed nmADMM algorithm achieves lowest estimation error under impulsive noise conditions as well as in AWGN environments at low SNR levels (≤ 10 dB).

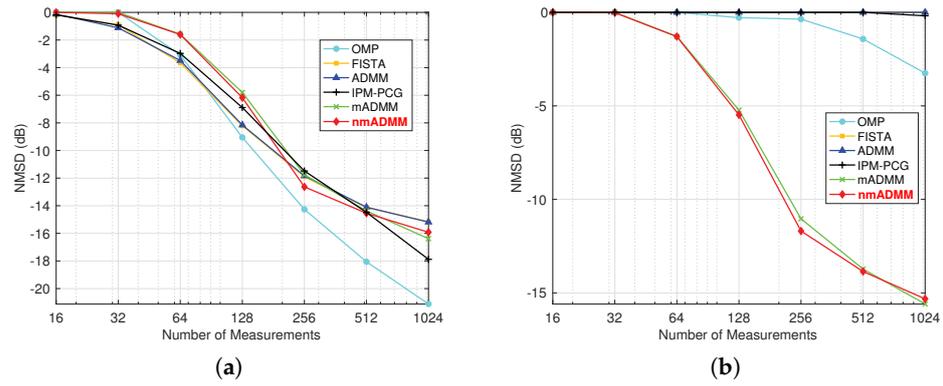


Figure 4. Variation in NMSD with different numbers of measurements. (a) AWGN environment. (b) Impulsive noise environment.

Table 1. Performance comparisons for CIR estimation in the single-carrier system with respect to the average number of iterations, runtime, and NMSD in AWGN (left side of '/') and impulsive noise (right side of '/') environments.

Method	# Iter.	Runtime (s)	NMSD (dB)
OMP [15]	64/64	0.32/0.32	−12.05/8.85
FISTA [23]	43/40	0.11/0.17	−11.89/7.85
ADMM [25]	35/23	0.12/0.11	−11.83/8.45
IPM-PCG [21]	24/21	11.86/24.66	−11.48/8.67
mADMM [28]	54/57	0.23/0.25	−12.20/−11.25
nmADMM	43/42	0.31/0.33	−12.22/−11.49

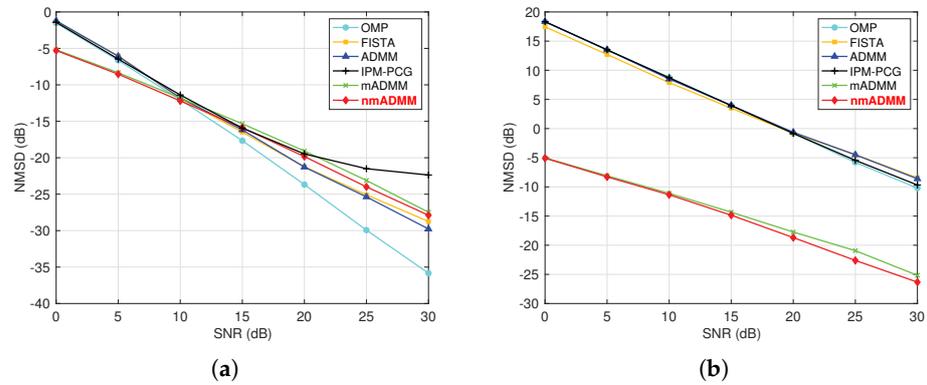


Figure 5. Variation in NMSD with different SNR levels. (a) AWGN environment. (b) Impulsive noise environment.

4.2. Simulation II: Doubly Spread Channel Estimation

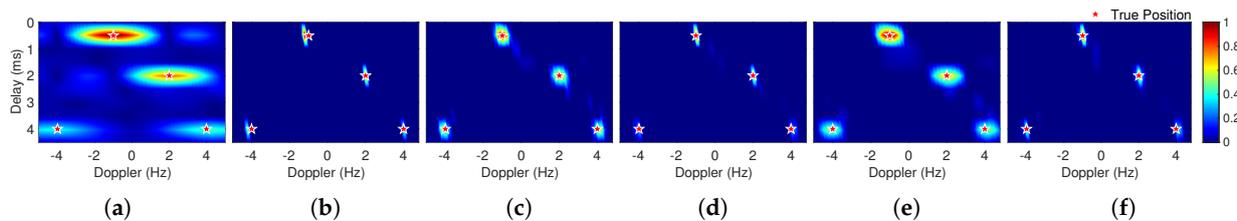
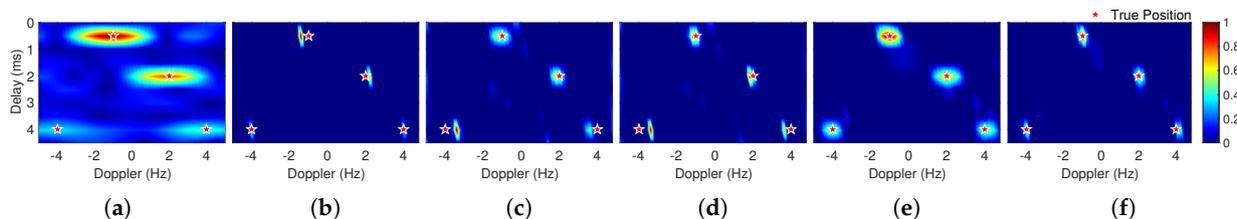
In this section, we simulate a four-path DD spread channel with the parameters detailed in Table 2. The SNR is set at 10 dB, aligning with the moderate noise level commonly encountered in practical UWA conditions. Impulsive noise probability is configured at $q = 10^{-3}$ with an INR of 30 dB. Similar to *Simulation I*, a PRBS employing BPSK modulation serves as the probe signal, featuring a symbol rate $R_s = 2$ kbaud and a pulse-shaping roll-off factor of 0.4. For channel estimation, we assign an upper bound for the maximum channel delay at $\tau_{\max} = 5$ ms, where the delay domain sampling interval matches the symbol duration at $\Delta\tau = 1/R_s = 0.5$ ms. The Doppler shift sampling interval is set at $\Delta\nu = 0.2$ Hz, with estimation bounds ranging from $\nu_{\min} = -5$ Hz to $\nu_{\max} = 5$ Hz. This leads to a total of $N = 500$ unknown channel parameters to estimate. For sparse channel estimation, the probe symbol sequence length is $M = \frac{2}{3}N \approx 330$, highlighting the problem is ill-posed. The second-order method, IPM-PCG, is excluded due to computational intensity and reduced accuracy in complex-valued problems.

Table 2. Parameter setting of the simulated DD channel.

Path No.	1	2	3	4
Delays (ms)	0.5	2	4	4
Doppler shift (Hz)	−1	2	−4	4
Modulus of amplitude	1	0.8	0.5	0.5

Before performing sparse methods, the Cross-Ambiguity Function (CAF) serves as our benchmark for channel spread function estimation [44]. The resolution of the CAF method is highly dependent on the auto-correlation properties of the probe signal. To effectively distinguish the four paths, we extend the length of the probe signal for CAF to $M = \frac{4}{3}N \approx 670$. The delay and Doppler resolutions approximate $\tau_{\text{res}} \approx \frac{1}{R_s} = 0.5$ ms and $\nu_{\text{res}} \approx \frac{R_s}{M} = 3$ Hz. Since the problem here has a similar scale to the CIR channel estimation problem in the previous section, we maintain consistent parameter settings for all algorithms except the stopping criteria. For ADMM-based methods, the absolute and relative tolerances are set at $\epsilon_{\text{abs}} = 5 \times 10^{-5}$ and $\epsilon_{\text{rel}} = 5 \times 10^{-4}$, respectively. The residue ratio tolerance for the FISTA algorithm is set at $\epsilon_{\text{fista}} = 10^{-6}$. We reduced the tolerance thresholds to achieve finer estimation precision, particularly as inaccuracies in the Doppler frequency shift estimation can severely impair subsequent equalization quality. The iteration number for the OMP algorithm is set at four, aligning with the sparsity of the simulated channel.

Figures 6 and 7 display the estimated DD spread functions under AWGN and impulsive noise environments. Following the probe signal, 2,000 data symbols of the same symbol rate and modulation configurations are transmitted through the channel. Based on the channel input–output relationships shown in (4)–(6), the estimated DD functions are first converted to the time-varying CIR, and then fed into the linear MMSE equalizer to retrieve the distorted data symbols. The equalization outputs using Ground Truth (GT) channel information and estimations from different methods are shown in Figures 8 and 9. We summarize the average number of iterations, runtime for single estimation, number of error bits, and Bit Error Rate (BER) of the data sequence in Table 3. Notice that given the limitations of the MMSE equalizer in addressing impulse noise, the BER in an impulsive noise environment rises by 8.75%, even when utilizing GT channel information.

**Figure 6.** DD channel spread function estimation in AWGN environment: (a) CAF, (b) OMP, (c) FISTA, (d) ADMM, (e) mADMM, and (f) nmADMM.**Figure 7.** DD channel spread function estimation in impulsive noise environment: (a) CAF, (b) OMP, (c) FISTA, (d) ADMM, (e) mADMM, and (f) nmADMM.

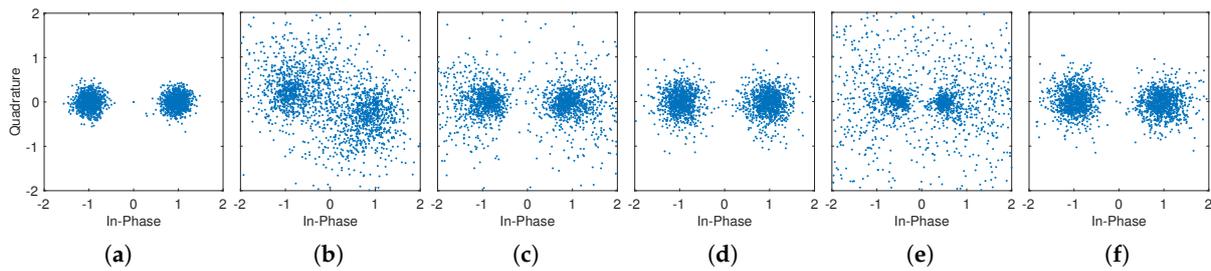


Figure 8. Constellation diagrams from linear MMSE equalizer outputs using different estimation results in AWGN environment: (a) GT, (b) OMP, (c) FISTA, (d) ADMM, (e) mADMM, (f) nmADMM.

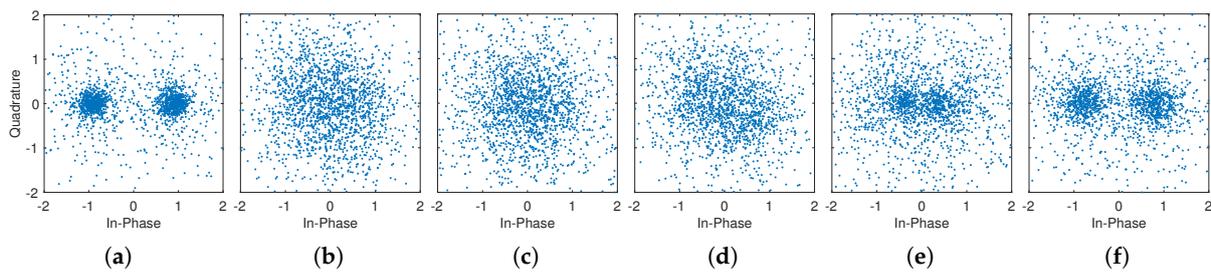


Figure 9. Constellation diagrams from linear MMSE equalizer outputs using different estimation results in impulsive noise environment: (a) GT, (b) OMP, (c) FISTA, (d) ADMM, (e) mADMM, (f) nmADMM.

Table 3. Performance comparison for DD spread function estimation in AWGN (left of '/') and impulsive noise (right of '/') environments with respect to average iterations, computing time, bit errors, and BER.

Method	# Iter.	Runtime (s)	# Err.	BER (%)
Ground Truth	-/-	-/-	0/179	0/8.75
OMP [15]	4/4	0.004/0.004	179/791	8.95/39.55
FISTA [23]	564/835	1.16/1.54	9/729	0.45/ 36.45
ADMM [25]	662/610	0.51/0.63	0/619	0/30.95
mADMM [28]	1577/2193	5.85/7.89	557/547	27.85/27.35
nmADMM	481/527	3.15/3.51	0/252	0/12.60

In comparison to the CIR estimation discussed in the previous section, the current problem exhibits a similar scale but features higher sparsity and a greater number of observations. However, all sparse algorithms demand more iterations, except for OMP. This is not merely due to the reduced tolerances for stopping criteria but also stems from the “poor” property of the measurement matrix Φ . In *Simulation 1*, the measurement matrix constructed directly by the PRBS had an average rank of 256 and a mutual coherence of 0.24. In contrast, for the problem here, the matrix determined by (8) has a rank of 140 and mutual coherence of 0.998. The high correlation among column vectors in Φ makes the identification of correct sparse solutions more difficult, particularly for greedy methods such as OMP. The iterative process of OMP, which selects the column most correlated with the current residue, can misidentify the correct support element due to this high coherence, further leading the entire estimate to be severely incorrect [19]. This is evident in Figure 8b, where the estimated channel components of OMP deviate from true values, leading to phase rotation during equalization even in the AWGN environment. Compared to greedy methods, convex optimization-based approaches such as FISTA, ADMM, and the proposed algorithm require less restrictive conditions on the measurement matrix for successful sparse estimation [45,46]. One can observe from Figure 6c–f that all convex optimization methods accurately identify the DD channel path in AWGN environment. However, under the influence of impulsive noise, FISTA and the original ADMM algorithms exhibit

increased estimation errors, particularly in identifying paths with smaller amplitudes. As illustrated in Figure 7c,d, the estimated locations of paths 3 and 4 by FISTA and the original ADMM algorithms deviate from their true positions. The high mutual coherence of the measurement matrix also makes the optimization landscape of non-differentiable $\ell_1 - \ell_1$ objective function more challenging, impacting both the convergence rate and the accuracy of the optimization algorithm. As shown in Figures 6 and 7, the DD functions estimated by the mADMM algorithm appear dispersive even after extensive iterations, although it retains a similar performance in both AWGN and impulsive noise settings. This is also evident in Figure 10, which illustrates the iteration curves of objective functions for both the mADMM and nmADMM algorithms. One can see that the accelerated gradient scheme and non-monotone line search strategy of the nmADMM algorithm facilitate faster convergence and greater precision. The proposed nmADMM algorithm can identify both the locations and amplitudes of the DD channel paths successfully, achieving the lowest BER with minimal iterations across both AWGN and impulsive noise environments.

Table 4 presents the comparison results of the BER performance of various algorithms at INR of 15 and 25 dB across different probabilities (q) of impulsive noise occurrence. The results demonstrate that the nmADMM algorithm achieves the lowest BER, closely approximating the BER derived from the GT channel information across all scenarios.

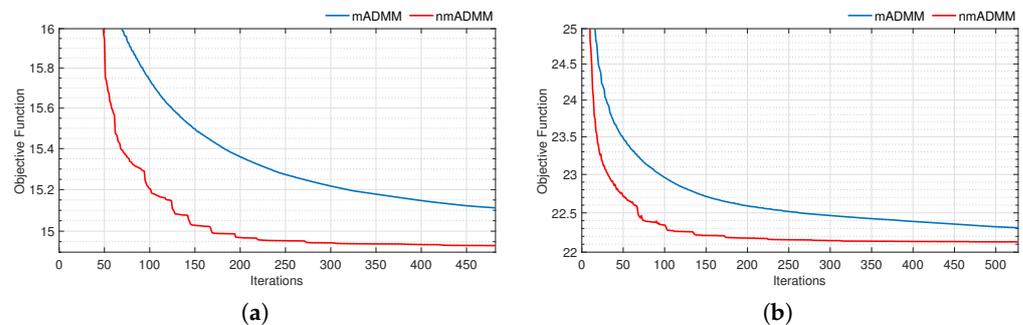


Figure 10. Iteration curves of the objective function for mADMM and nmADMM in DD channel estimation: (a) AWGN environment and (b) impulsive noise environment.

Table 4. Comparison of BER performance across different algorithms under INR conditions of 15 and 25 dB with impulsive noise occurrence probabilities of $q = 0.01, 0.005, \text{ and } 0.001$.

q	INR = 15 dB			INR = 25 dB		
	0.01	0.005	0.001	0.01	0.005	0.001
Ground Truth	2.10	0.80	0.30	18.80	8.85	2.50
OMP [15]	36.25	34.95	10.25	48.60	47.20	35.85
FISTA [23]	27.85	11.65	0.50	55.35	45.80	8.35
ADMM [25]	27.65	13.55	0.55	55.25	42.85	8.10
mADMM [28]	5.70	4.30	3.35	23.90	15.15	8.30
nmADMM	5.25	2.05	0.45	22.15	13.50	3.85

5. At-Sea Experiment

To evaluate the real-world performance of the proposed nmADMM algorithm for UWA channel estimation, we conducted an at-sea experiment. The data were collected in the shallow waters of Wuyuan Bay, Xiamen, China, in July 2021 (GPS: $24^{\circ}32'10''\text{N}$, $118^{\circ}11'31''\text{E}$). Figure 11a depicts the location and deployment of the experiment. The transmitter and receiver, separated by approximately 1 km, were both anchored 2 m deep near the coastline. During the experiment, CP-OFDM signals with subcarriers modulated by Quadrature Phase Shift Keying (QPSK) symbols were transmitted. The center frequency of the system is 21 kHz, with a sampling frequency of 96 kHz. In the OFDM configuration, 256 subcarriers with 23.4 Hz spacing, an FFT size of 4096, and a CP length of 3840 were utilized. Each OFDM symbol had a length of 82.7 ms. Figure 11b presents the frame

format of the transmitted signal, where each frame of the signal consists of one Hyperbolic Frequency Modulation (HFM) preamble signal followed by OFDM symbols. A received signal sample is shown in Figure 12a. As the experiment site is near the coast, various industrial and human activities and the flow noise resulting from water currents and turbulence make the received signal contaminated by strong impulsive noise. Before channel estimation, pre-processing is conducted, including large impulsive noise reduction via peak detection and threshold clipping. Figure 12b shows the signal sample after removing large impulsive noise. Following frame synchronization through matched filtering with HFM preamble signals, large-scale Doppler spread is compensated for by resampling. The Carrier Frequency Offset (CFO) resulting from residual Doppler effects is estimated and mitigated through a one-dimensional search strategy, as detailed in [47]. The pre-processed signal is then forwarded to various channel estimators for evaluations. Consistent with the hardware configurations employed in previous numerical simulations, all algorithms are run on MATLAB 2023a, Windows 10, using a Dell-manufactured desktop PC with an Intel Core i5 2.30 GHz CPU and 16 GB RAM.

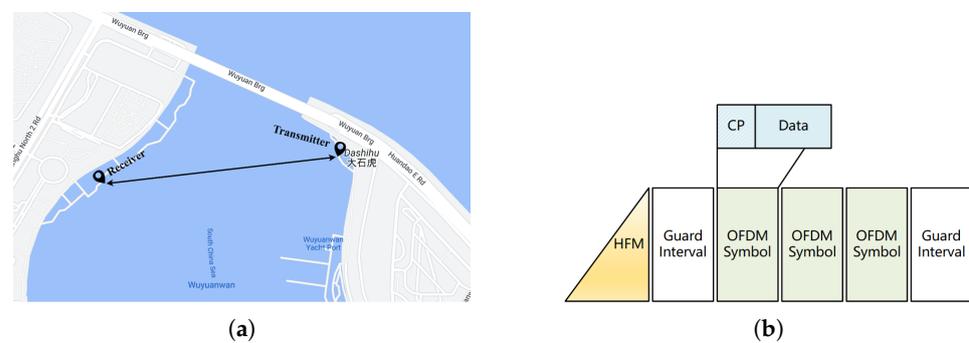


Figure 11. Experiment configuration: (a) Experiment site map at Wuyuan Bay, Xiamen City, China, and (b) frame format of the transmitted signal.

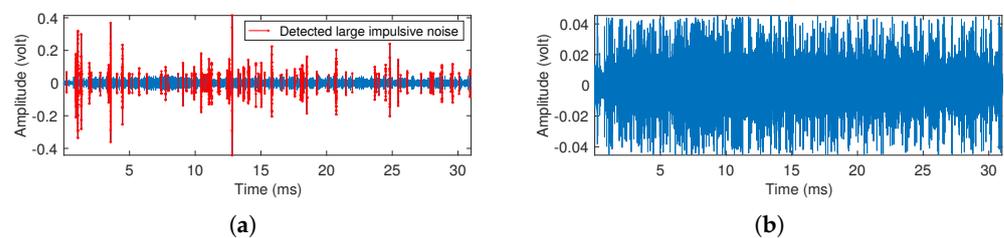


Figure 12. Sample of the received signal: (a) Raw signal and (b) signal after pre-processing to eliminate large impulsive noise.

We adopted block-type, pilot-based channel estimation using the first OFDM symbol of each frame. The channel order is set to the maximum value that the estimator can take, which corresponds to the length of the CP (i.e., $N = 3840$). Pilot symbols carried by $M = 256$ subcarriers are used for channel estimation, where $M \ll N$ indicates the problem is highly ill-defined. In addition to classic OMP, FISTA, and ADMM algorithms, we incorporate recently introduced channel estimation methods for OFDM systems, including the Adaptive OMP (A-OMP) algorithm [14] and the Fast Marginal Likelihood Maximization (FM) based Sparse Bayesian Learning (SBL) algorithm [48], for comparative analysis. Figure 13 presents the estimated channel via different algorithms. The monotone strategy-based mADMM algorithm is not included here due to its slow convergence rate with high mutual coherence measurement matrices, as demonstrated in Section 4.2. Regularization factors for FISTA and ADMM were set to $\lambda = 0.01 \lambda_{\infty}$, and for the nmADMM, $\tau = 1/(0.02\lambda_{\infty})$. We configured the tolerance parameters of the stopping criteria to $\epsilon_{\text{abs}} = 10^{-2}$, $\epsilon_{\text{rel}} = 10^{-1}$, and $\epsilon_{\text{fista}} = 10^{-3}$. The convergence tolerance parameter for the FM-SBL algorithm is set to $\epsilon_{\text{fm-sbl}} = 10^{-4}$. The prior sparsity information required by the OMP algorithm is derived from coarse channel estimation using matched filtering on the preamble signal. The

A-OMP algorithm does not require prior sparsity information for the channel. We set its stopping criterion, the power ratio of the residual to the received symbols, at 5×10^{-2} through empirical evaluations.

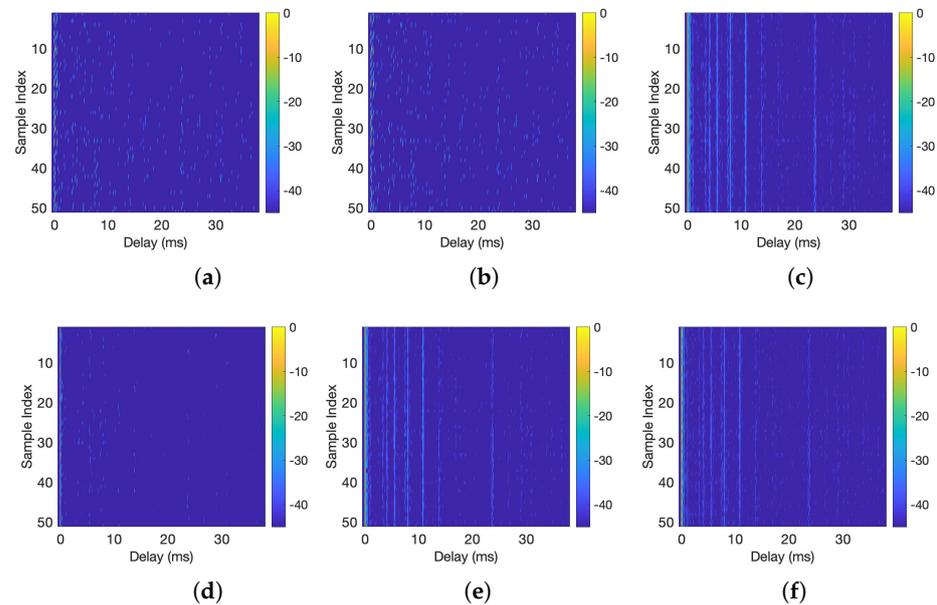


Figure 13. Estimated CIR from at-sea experiment using (a) OMP, (b) A-OMP, (c) FISTA, (d) FM-SBL, (e) ADMM, and (f) nmADMM.

After the channel estimation, the derived results are utilized to restore the subsequent distorted data symbols using the linear MMSE equalizer. Figure 14 presents the constellation diagrams of the equalized QPSK symbols based on channel estimation from different algorithms. A total of 51 signal frames with 26,112 received QPSK data symbols were analyzed. Table 5 shows a comparison in terms of the average number of iterations, computation time, and Symbol Error Rate (SER) for the four algorithms. It can be observed that the greedy method OMP has the lowest computational complexity but with low estimation accuracy and high SER of 10.68%. The A-OMP algorithm, with an improved termination condition, lowers the SER by 3.23% compared to the OMP algorithm. When contrasted against numerical simulations of smaller scale in the previous section, the computational cost of the original ADMM algorithm increases sharply due to the evaluation of matrix inverse $(\Phi^H \Phi + \rho \mathbf{I})^{-1}$ in each estimation. Although the FM-SBL algorithm has reduced computational complexity of less than $\mathcal{O}(M^2 N)$ compared to the Expectation Maximization (EM)-SBL algorithm [48], it exhibits the second highest runtime at 9.73 s. The advantages of the proposed nmADMM algorithm are less pronounced compared to the simulations, as the time-to-frequency domain conversion in OFDM demodulation and the pre-processing mitigate the effects of impulsive noise. Nevertheless, it maintains high estimation accuracy, delivering the lowest SER of 4.48%. Figure 15 examines the convergence behavior of the nmADMM algorithm, showcasing iteration curves of the objective function and both primal and dual residues. One can see that the proposed algorithm can converge within tens of iterations in all estimations.

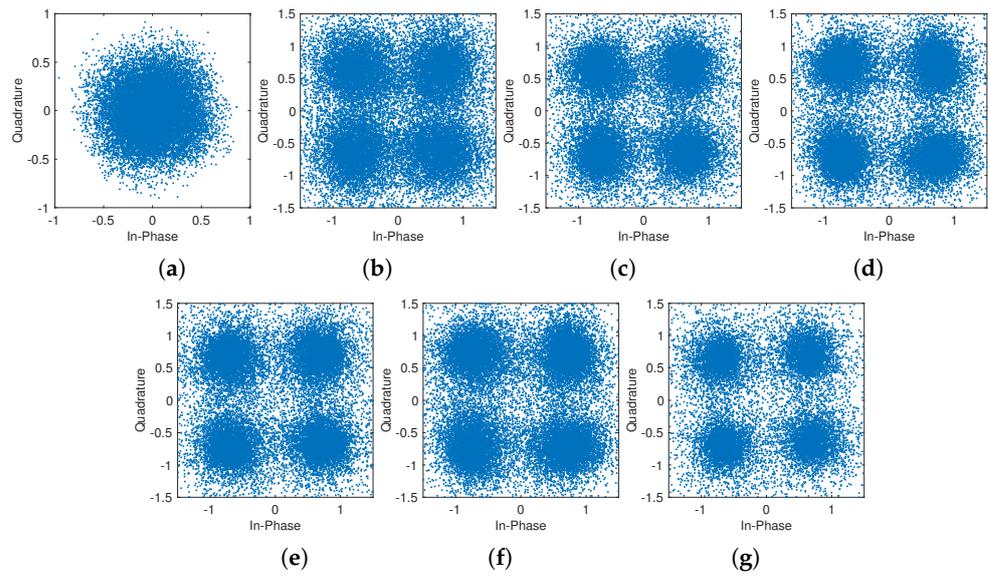


Figure 14. Constellation diagrams of raw received symbols and equalization outputs based on various channel estimates: (a) raw received symbols, (b) OMP, (c) A-OMP, (d) FISTA, (e) FM-SBL, (f) ADMM, and (g) nmADMM.

Table 5. Performance comparison for channel estimation in the at-sea experiment including average number of iterations, runtime for single estimation, and SER.

Method	# Iter.	Runtime (s)	SER (%)
OMP [15]	74	0.3152	10.68
A-OMP [14]	85	0.56	7.45
FISTA [23]	24	1.0852	5.87
FM-SBL [48]	57	9.73	5.83
ADMM [25]	47	13.1543	6.61
nmADMM	32	1.4681	4.48

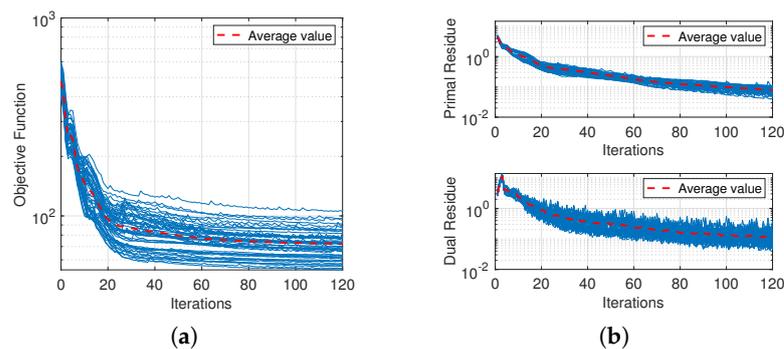


Figure 15. Convergence analysis of the nmADMM algorithm: Iteration curves of (a) the objective function and (b) the primal residue (top) and the dual residue (bottom).

6. Conclusions

In this work, we present a robust channel estimation method for UAC in impulsive noise environments. Our approach redefines channel estimation as an $\ell_1 - \ell_1$ optimization problem, leveraging UWA channel sparsity and addressing impulsive noise outliers. Using the ADMM framework, we decompose this convex but non-differentiable problem into two subproblems, each efficiently solved using the APG method. We also introduce a non-monotone backtracking line search strategy to adaptively adjust the step-size parameter, avoiding the need for complex evaluation of the Lipschitz constant.

The proposed nmADMM algorithm, with a complexity of $\mathcal{O}(MN)$ and devoid of matrix inversion or multiplication operations, is tested across various estimation tasks and communication schemes. Numerical simulations show that only a slight NMSD increase of 0.73 dB occurs in impulsive noise environments, compared to AWGN conditions, at an SNR of 10 dB and an INR of 30 dB for CIR estimation. Compared with its monotone counterpart, the mADMM algorithm, the proposed algorithm demonstrates faster convergence speed and higher estimation accuracy for both CIR and DD spread function estimations across various SNR and INR levels. At-sea experiments further validate that the nmADMM algorithm outperforms the OMP, A-OMP, FISTA, ADMM, and FM-SBL algorithms, achieving the lowest raw SER with simple MMSE equalization at 4.48%. Consistently, the nmADMM algorithm converges within tens of iterations, proving its robustness in both AWGN and impulsive noise environments.

Author Contributions: Conceptualization, T.T., K.Y. and F.-Y.W.; methodology, T.T., K.Y. and F.-Y.W.; software, T.T.; validation, T.T., K.Y., F.-Y.W. and Y.Z.; formal analysis, T.T. and Y.Z.; investigation, T.T. and F.-Y.W.; resources, K.Y. and F.-Y.W.; data curation, T.T.; writing—original draft preparation, T.T.; supervision, K.Y., F.-Y.W. and Y.Z.; project administration, K.Y.; funding acquisition, K.Y. and F.-Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 62171369 and in part by the Key Program of the National Natural Science Foundation of China under Grant 52231013.

Data Availability Statement: Dataset will be made available on request from the authors.

Acknowledgments: The authors would like to thank Shengxing Liu and his team at the College of Ocean and Earth Science, Xiamen University, for their invaluable assistance in setting up the experiment at Wuyuan Bay.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Jahanbakht, M.; Xiang, W.; Hanzo, L.; Rahimi Azghadi, M. Internet of Underwater Things and Big Marine Data Analytics—A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 904–956. [\[CrossRef\]](#)
- Zhou, Y.; Wang, R.; Yang, X.; Tong, F. Orthogonal Projection and Distributed Compressed Sensing-Based Impulsive Noise Estimation for Underwater Acoustic OSDM Communication. *IEEE Internet Things J.* **2023**, *10*, 22279–22293. [\[CrossRef\]](#)
- Bello, O.; Zeadally, S. Internet of Underwater Things Communication: Architecture, Technologies, Research Challenges and Future Opportunities. *Ad Hoc Netw.* **2022**, *135*, 102933. [\[CrossRef\]](#)
- Wei, X.; Guo, H.; Wang, X.; Wang, X.; Qiu, M. Reliable Data Collection Techniques in Underwater Wireless Sensor Networks: A Survey. *IEEE Commun. Surv. Tutor.* **2022**, *24*, 404–431. [\[CrossRef\]](#)
- Ouyang, D.; Li, Y.; Wang, Z. Channel Estimation for Underwater Acoustic OFDM Communications: An Image Super-Resolution Approach. In Proceedings of the ICC 2021—IEEE International Conference on Communications, Montreal, QC, Canada, 14–23 June 2021; pp. 1–6.
- Huang, S.H.; Tsao, J.; Yang, T.C.; Cheng, S.W. Model-Based Signal Subspace Channel Tracking for Correlated Underwater Acoustic Communication Channels. *IEEE J. Ocean. Eng.* **2014**, *39*, 343–356. [\[CrossRef\]](#)
- Wu, F.Y.; Tian, T.; Su, B.X.; Song, Y.C. Hadamard–Viterbi Joint Soft Decoding for MFSK Underwater Acoustic Communications. *Remote Sens.* **2022**, *14*, 6038. [\[CrossRef\]](#)
- Kaddouri, S.; Beaujean, P.P.J.; Bouvet, P.J.; Real, G. Least Square and Trended Doppler Estimation in Fading Channel for High-Frequency Underwater Acoustic Communications. *IEEE J. Ocean. Eng.* **2014**, *39*, 179–188. [\[CrossRef\]](#)
- Athaudage, C.; Jayalath, A. Enhanced MMSE Channel Estimation Using Timing Error Statistics for Wireless OFDM Systems. *IEEE Trans. Broadcast.* **2004**, *50*, 369–376. [\[CrossRef\]](#)
- Panayirci, E.; Altabbaa, M.T.; Uysal, M.; Poor, H.V. Sparse Channel Estimation for OFDM-Based Underwater Acoustic Systems in Rician Fading with a New OMP-MAP Algorithm. *IEEE Trans. Signal Process.* **2019**, *67*, 1550–1565. [\[CrossRef\]](#)
- Wu, F.Y.; Yang, K.; Tong, F.; Tian, T. Compressed Sensing of Delay and Doppler Spreading in Underwater Acoustic Channels. *IEEE Access* **2018**, *6*, 36031–36038. [\[CrossRef\]](#)
- Li, W.; Preisig, J.C. Estimation of Rapidly Time-Varying Sparse Channels. *IEEE J. Ocean. Eng.* **2007**, *32*, 927–939. [\[CrossRef\]](#)
- Sun, Q.; Wu, F.Y.; Yang, K.; Ma, Y. Estimation of Multipath Delay-Doppler Parameters from Moving LFM Signals in Shallow Water. *Ocean Eng.* **2021**, *232*, 109125. [\[CrossRef\]](#)
- Wang, Z.; Li, Y.; Wang, C.; Ouyang, D.; Huang, Y. A-OMP: An Adaptive OMP Algorithm for Underwater Acoustic OFDM Channel Estimation. *IEEE Wirel. Commun. Lett.* **2021**, *10*, 1761–1765. [\[CrossRef\]](#)

15. Cai, T.T.; Wang, L. Orthogonal Matching Pursuit for Sparse Signal Recovery With Noise. *IEEE Trans. Inf. Theory* **2011**, *57*, 4680–4688. [[CrossRef](#)]
16. Wu, F.Y.; Song, Y.C.; Yang, K. An Effective Framework for Underwater Acoustic Data Acquisition. *Appl. Acoust.* **2021**, *182*, 108235. [[CrossRef](#)]
17. Donoho, D.L.; Tsai, Y.; Drori, I.; Starck, J.L. Sparse Solution of Underdetermined Systems of Linear Equations by Stagewise Orthogonal Matching Pursuit. *IEEE Trans. Inf. Theory* **2012**, *58*, 1094–1121. [[CrossRef](#)]
18. Fletcher, A.K.; Rangan, S.; Goyal, V.K. Necessary and Sufficient Conditions for Sparsity Pattern Recovery. *IEEE Trans. Inf. Theory* **2009**, *55*, 5758–5772. [[CrossRef](#)]
19. Ben-Haim, Z.; Eldar, Y.C.; Elad, M. Coherence-Based Performance Guarantees for Estimating a Sparse Vector under Random Noise. *IEEE Trans. Signal Process.* **2010**, *58*, 5030–5043. [[CrossRef](#)]
20. Berger, C.R.; Zhou, S.; Preisig, J.C.; Willett, P. Sparse Channel Estimation for Multicarrier Underwater Acoustic Communication: From Subspace Methods to Compressed Sensing. *IEEE Trans. Signal Process.* **2010**, *58*, 1708–1721. [[CrossRef](#)]
21. Kim, S.J.; Koh, K.; Lustig, M.; Boyd, S.; Gorinevsky, D. An Interior-Point Method for Large-Scale-Regularized Least Squares. *IEEE J. Sel. Top. Signal Process.* **2007**, *1*, 606–617. [[CrossRef](#)]
22. Zheng, P.; Lyu, X.; Gong, Y. Trainable Proximal Gradient Descent-Based Channel Estimation for mmWave Massive MIMO Systems. *IEEE Wirel. Commun. Lett.* **2023**, *12*, 1781–1785. [[CrossRef](#)]
23. Beck, A.; Teboulle, M. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM J. Imaging Sci.* **2009**, *2*, 183–202. [[CrossRef](#)]
24. Liu, H.; Zhang, J.; Wu, Q.; Xiao, H.; Ai, B. ADMM Based Channel Estimation for RISs Aided Millimeter Wave Communications. *IEEE Commun. Lett.* **2021**, *25*, 2894–2898. [[CrossRef](#)]
25. Boyd, S. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* **2010**, *3*, 1–122. [[CrossRef](#)]
26. Khan, M.R.; Das, B.; Pati, B.B. Channel Estimation Strategies for Underwater Acoustic (UWA) Communication: An Overview. *J. Frankl. Inst.* **2020**, *357*, 7229–7265. [[CrossRef](#)]
27. Kuai, X.; Sun, H.; Zhou, S.; Cheng, E. Impulsive Noise Mitigation in Underwater Acoustic OFDM Systems. *IEEE Trans. Veh. Technol.* **2016**, *65*, 8190–8202. [[CrossRef](#)]
28. Tian, T.; Raj, A.; Xavier, B.M.; Zhang, Y.; Wu, F.Y.; Yang, K. A Robust ADMM-Based Optimization Algorithm For Underwater Acoustic Channel Estimation. In Proceedings of the OCEANS 2023—Limerick IEEE Conference, Limerick, Ireland, 5–8 June 2023; pp. 1–5.
29. Ye, H.; Li, G.Y.; Juang, B.H. Power of Deep Learning for Channel Estimation and Signal Detection in OFDM Systems. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 114–117. [[CrossRef](#)]
30. Jiang, R.; Wang, X.; Cao, S.; Zhao, J.; Li, X. Deep Neural Networks for Channel Estimation in Underwater Acoustic OFDM Systems. *IEEE Access* **2019**, *7*, 23579–23594. [[CrossRef](#)]
31. Liu, L.; Cai, L.; Ma, L.; Qiao, G. Channel State Information Prediction for Adaptive Underwater Acoustic Downlink OFDMA System: Deep Neural Networks Based Approach. *IEEE Trans. Veh. Technol.* **2021**, *70*, 9063–9076. [[CrossRef](#)]
32. Lv, X.; Li, Y.; Wu, Y.; Wang, X.; Liang, H. Joint Channel Estimation and Impulsive Noise Mitigation Method for OFDM Systems Using Sparse Bayesian Learning. *IEEE Access* **2019**, *7*, 74500–74510. [[CrossRef](#)]
33. Chen, P.; Rong, Y.; Nordholm, S.; He, Z. Joint Channel and Impulsive Noise Estimation in Underwater Acoustic OFDM Systems. *IEEE Trans. Veh. Technol.* **2017**, *66*, 10567–10571. [[CrossRef](#)]
34. Lee, J.; Park, C.; Ryu, E. A Geometric Structure of Acceleration and Its Role in Making Gradients Small Fast. *Neural Inf. Process. Syst.* **2021**, *34*, 11999–12012.
35. Zhang, Y.; Venkatesan, R.; Dobre, O.A.; Li, C. Efficient Estimation and Prediction for Sparse Time-Varying Underwater Acoustic Channels. *IEEE J. Ocean. Eng.* **2020**, *45*, 1112–1125. [[CrossRef](#)]
36. Parikh, N.; Boyd, S. Proximal Algorithms. *Found. Trends Optim.* **2014**, *1*, 127–239. [[CrossRef](#)]
37. Wen, F.; Liu, P.; Liu, Y.; Qiu, R.C.; Yu, W. Robust Sparse Recovery in Impulsive Noise via ℓ_p - ℓ_1 Optimization. *IEEE Trans. Signal Process.* **2017**, *65*, 105–118. [[CrossRef](#)]
38. Borsic, A.; Adler, A. A Primal–Dual Interior-Point Framework for Using the L1 or L2 Norm on the Data and Regularization Terms of Inverse Problems. *Inverse Probl.* **2012**, *28*, 095011. [[CrossRef](#)]
39. Wang, S.; Liu, Q.; Xia, Y.; Dong, P.; Luo, J.; Huang, Q.; Feng, D.D. Dictionary Learning Based Impulse Noise Removal via L1–L1 Minimization. *Signal Process.* **2013**, *93*, 2696–2708. [[CrossRef](#)]
40. Jiang, J.; Wang, Z.; Chen, C.; Lu, T. L1-L1 Norms for Face Super-Resolution with Mixed Gaussian-impulse Noise. In Proceedings of the 41st IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016), Shanghai, China, 20–25 March 2016; pp. 2089–2093.
41. Zhang, H.; Hager, W.W. A Nonmonotone Line Search Technique and Its Application to Unconstrained Optimization. *SIAM J. Optim.* **2004**, *14*, 1043–1056. [[CrossRef](#)]
42. Li, H.; Lin, Z. Accelerated Proximal Gradient Methods for Nonconvex Programming. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), Montréal, QC, Canada, 7–12 December 2015; Volume 28.
43. Zhou, S.; Wang, Z. *OFDM for Underwater Acoustic Communications*; Wiley: Hoboken, NJ, USA, 2014.
44. Zeng, W.J.; Xu, W. Fast Estimation of Sparse Doubly Spread Acoustic Channels. *J. Acoust. Soc. Amer.* **2012**, *131*, 303–317. [[CrossRef](#)]

45. Emadi, M.; Miandji, E.; Unger, J. A Performance Guarantee for Orthogonal Matching Pursuit Using Mutual Coherence. *Circuits Syst. Signal Process.* **2018**, *37*, 1562–1574. [[CrossRef](#)]
46. Donoho, D.L.; Elad, M. Optimally Sparse Representation in General (Nonorthogonal) Dictionaries via ℓ_1 Minimization. *Proc. Natl. Acad. Sci. USA* **2003**, *100*, 2197–2202. [[CrossRef](#)]
47. Li, B.; Zhou, S.; Stojanovic, M.; Freitag, L.; Willett, P. Multicarrier Communication Over Underwater Acoustic Channels with Nonuniform Doppler Shifts. *IEEE J. Ocean. Eng.* **2008**, *33*, 198–209.
48. Jia, S.; Zou, S.; Zhang, X.; Da, L. Underwater Acoustic Channel Estimation Based on Sparse Bayesian Learning Algorithm. *IEEE Access* **2023**, *11*, 7829–7836. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.