Journal of
Marine Science
and Engineering

MDPI

Article

PN Sequence Period Estimation Method for Underwater Direct-Sequence Spread Spectrum Signals Under Low SNR

Yuanxin Teng [1,2,3], Yilin Wang [1,2,3], Yu Xiang [4], Jinjin Wang [1,2,3,*], Guolong Liang [1,2,3], Yu Hao [1,2,3] and Jin Fu [5]

1 National Key Laboratory of Underwater Acoustic Technology, Harbin Engineering University, Harbin 150001, China; tyxlovely@hrbeu.edu.cn (Y.T.); wangyilin@hrbeu.edu.cn (Y.W.); liangguolong@hrbeu.edu.cn (G.L.); haoyu@hrbeu.edu.cn (Y.H.)
2 Key Laboratory of Marine Information Acquisition and Security (Harbin Engineering University), Ministry of Industry and Information Technology, Harbin 150001, China
3 College of Underwater Acoustic Engineering, Harbin Engineering University, Harbin 150001, China
4 Wuhan Second Ship Design and Research Institute, Wuhan 430064, China; xiangyu8899@hrbeu.edu.cn
5 School of Electronic Information Engineering, Beihang University, Beijing 100094, China; fujin@hrbeu.edu.cn
* Correspondence: wangjinjin@hrbeu.edu.cn

Abstract

In non-cooperative DSSS signal reception, the accurate estimation of the pseudo-noise (PN) sequence period is essential for successful despreading and information recovery. In this paper, we propose an average second-order-moment autocorrelation method based on the accumulated code difference function to enhance the estimation accuracy under low-signal-to-noise-ratio (SNR) conditions. The proposed approach involves three key aspects: First, a quadrature receiver is employed to mitigate the impact of the random initial phase on demodulation, enabling the full utilization of the signal energy. Second, a code difference function is constructed using transition information between adjacent spreading codes, and by leveraging the strong correlation between these functions, accumulated processing effectively suppresses the noise influence. Third, the autocorrelation result of the accumulated code difference function displays periodic peaks separated by intervals equal to the PN sequence period, allowing for period estimation through peak interval extraction. In addition, the introduced average-second-order-moment technique addresses the peak loss caused by information code randomness while further smoothing noise. Simulation and experimental results verify the effectiveness and practicality of the proposed method, which outperformed the average-second-order-moment method by about 1.5 dB and the accumulated-power-spectrum-reprocessing method by about 2 dB.

Keywords: direct-sequence spread spectrum (DSSS) signals; pseudo-noise (PN) sequence period estimation; code difference function; autocorrelation

check for updates

Academic Editor: Weicheng Cui

Received: 12 November 2025
Revised: 3 December 2025
Accepted: 4 December 2025
Published: 6 December 2025

Citation: Teng, Y.; Wang, Y.; Xiang, Y.; Wang, J.; Liang, G.; Hao, Y.; Fu, J. PN Sequence Period Estimation Method for Underwater Direct-Sequence Spread Spectrum Signals Under Low SNR. *J. Mar. Sci. Eng.* **2025**, *13*, 2318. https://doi.org/10.3390/jmse13122318

1. Introduction

The human exploration of marine resources is increasing, driving significant advancements in underwater acoustic communication technology [1]. Direct-sequence spread spectrum (DSSS) signals, known for their strong anti-jamming capabilities and excellent concealment features [2–4], are widely used in both civilian and military underwater communication systems [5–8], and their reconnaissance has also become essential for ensuring maritime security. However, accurately decoding these signals poses a significant challenge in non-cooperative communication systems, as the receiver has no prior knowledge of the DSSS signal parameters used by the transmitter [9]; thus, in such scenarios, the estimation of the DSSS signal parameters is critical [10,11], especially the key parameters, which include the carrier frequency, chip width, and pseudo-noise (PN) sequence period. In recent years, the estimation methods for the carrier frequency and chip width have become relatively mature. However, achieving the high-precision period estimation of PN sequences under low-signal-to-noise-ratio (SNR) conditions remains challenging, a limitation that impedes the accurate estimation of the subsequent PN sequence, thereby affecting the decoding accuracy. Consequently, the precise estimation of PN sequence periods under low-SNR conditions has become a key research focus [12–15].

Common methods for the estimation of PN sequence periods include autocorrelation, power spectrum reprocessing, fourth-order cumulants, delay multiplication, and cepstrum and neural networks. Burel, G. et al. [16] proposed the average autocorrelation second-order moment to characterize fluctuations in the autocorrelation function. By leveraging the distinct fluctuation characteristics between noise and DSSS signals, their method enables DSSS signal detection and PN code period estimation through peak extraction. However, the performance of this method degrades significantly under low-SNR conditions. Building on this concept, Zhang Tianqi [17] proposed an autocorrelation estimator capable of simultaneously estimating the pseudocode period and chip rate, demonstrating that multiple DSSS signal parameters can be extracted through the autocorrelation with different integration times. Despite this advancement, the method offers limited improvement in period estimation under low-SNR conditions. In a separate study, Zhang Tianqi [18] proposed a power-spectrum-reprocessing technique, in which secondary spectral processing concentrates DSSS signal energy into sharp triangular pulses with intervals that correspond to the PN sequence period. While ensemble averaging in this method suppresses noise and mitigates information code randomness, its performance under low-SNR conditions remains inadequate. To address noise sensitivity, Zhang Yu et al. [19] incorporated wavelet decomposition as a preprocessing step into the accumulated second-order power spectrum method, enhancing the noise robustness; however, the performance heavily depends on the appropriate wavelet parameter selection. Shen Zhenhui et al. [20], leveraging the capability of higher-order cumulants to suppress Gaussian white noise, proposed an algorithm for PN sequence period estimation based on fourth-order-cumulant two-dimensional slicing. This method effectively reduces the computational complexity of higher-order cumulants and enables joint carrier frequency estimation, though further enhancements are needed for lower-SNR conditions. Kuehls, J. F. et al. [21] analyzed the periodicity exhibited by DSSS signals following delay-and-multiply processing, contributing another perspective to period estimation techniques. Building on this, Dong Zhanqi et al. [22] proposed a delay multiplication–correlation and spectrum analysis method, which not only allows for the estimation of the PN sequence and its period but also for the determination of the synchronization information [14]. Subsequently, Zhao Xiaolin et al. [23] introduced a delay multiplication–correlation method with wavelet decomposition preprocessing, which mitigates noise through wavelet decomposition. However, the performances of these algorithms are influenced by fixed delay values, and they generally require a coarse estimation of the chip rate to achieve high-precision PN sequence period estimation. When the fixed delay value is not optimal, their estimation performances drop sharply under low-SNR conditions. Zhong Sihui et al. introduced a method based on an improved cepstral spectrum [24] that achieves noise suppression through delay multiplication and then utilizes the high resolution of the Yule Walker autoregressive power spectrum in the modern power spectrum to optimize the cepstral spectrum, resulting in distinct, discrete spectral lines in the cepstral spectrum, thereby achieving high-precision pseudocode period estimation under low-SNR conditions. However, because the delay multiplication method is sensitive to noise suppression, the performance of this method is compromised at low SNRs when the delay is not optimal. Gu Hanqing [14] proposed a method based on ResNet that classifies PN sequences according to their lengths to achieve PN period estimation. However, this method relies on a large amount of training data, and for scenarios with substantial real-time requirements, the measured data are scarce, resulting in the poor generalization ability of the classification model and, ultimately, a decline in the classification performance.

To achieve the high-precision estimation of the PN sequence period for DSSS signals under low-SNR conditions, we treated the estimations of the carrier frequency and chip width of DSSS signals as preprocessing steps in this study, and we propose an estimation method based on the autocorrelation properties of the accumulated code difference function. The proposed approach involves three key aspects: First, a quadrature receiver is employed to mitigate the impact of the random initial phase on demodulation, enabling the full utilization of the signal energy. Second, a code difference function is constructed using transition information between adjacent spreading codes, and by leveraging the strong correlation between these functions, accumulated processing effectively suppresses the noise influence. Third, the autocorrelation result of the accumulated code difference function displays periodic peaks separated by intervals equal to the PN sequence period, allowing for period estimation through peak interval extraction. In addition, the introduced average-second-order-moment technique addresses the peak loss caused by information code randomness while further smoothing noise.

This paper is organized as follows: We present the research background and significance in Section 1. In Section 2, we describe the DSSS/BPSK signal model, which is demodulated using an orthogonal receiver, introduce the construction of the code difference function, and theoretically derive the output SNR expression for the accumulated code difference function, with simulations validating the theoretical results. In this section, we further analyze the autocorrelation characteristics of the accumulated code difference function and the impact of multipath effects on it, then provide the expression for its average second-order moment (ASOMA-ACDF). We detail the implementation procedure in Section 3, evaluate the estimation performance through simulations in Section 4, verify the method using measured data in Section 5, and conclude the paper in Section 6.

2. Method and Principle

2.1. Signal Modeling

Underwater communication channels have both multipath and Doppler spread. However, the Doppler spread broadening range during the observation time is usually much smaller than the bandwidth of the spread spectrum signal, and it is not the main factor that influences PN code period estimation. Therefore, we represent the Doppler effect in underwater acoustic channels as frequency shift in this article.

The received DSSS/BPSK signal should be represented as follows:

$$\begin{aligned} x(nT_s) &= \sum_{z=1}^{Z_0} x(nT_s - n_zT_s) = \sum_{z=1}^{Z_0} A_z c(nT_s - n_zT_s) p(nT_s - n_zT_s)\cos(\omega_0 nT_s + \varphi_z) + n_0(nT_s) \\ &= \sum_{z=1}^{Z_0} A_z d(nT_s - n_zT_s)\cos(\omega_0 nT_s + \varphi_z) + n_0(nT_s) \end{aligned} \tag{1}$$

where ω_0 is the received frequency after Doppler shift; Z_0 is the number of multipath paths; n is the sampling sequence number; T_s is the sampling interval; n_zT_s is the time delay for each path; A_z is the amplitude of the received signal for each path; φ_z is the initial phase of the received signal for each path; $n_0(nT_s)$ is the background Gaussian white noise; $p(nT_s) = \sum_{q=-\infty}^{\infty} p_q g_p(nT_s - qN_0T_s)$, $p_q \in \{-1, +1\}$ represents the source information sequence, where N_0T_s is the source information symbol duration; $g_p(nT_s)$ is a rectangular pulse spanning N_0T_s. The PN sequence is given by

$c(nT_s) = \sum_{m=-\infty}^{\infty} c_m g(nT_s - mN_cT_s)$, $c_m \in \{-1,+1\}$, where N_cT_s is the chip width of one code of the PN sequence. Here, $g(nT_s)$ is a rectangular pulse with a width of N_cT_s, $d(nT_s) = \sum_{i=-\infty}^{\infty} d_i g(nT_s - iN_cT_s)$, and $d_i \in \{-1,+1\}$ represents the spread spectrum information sequence.

To mitigate the effect of the random initial phase of the received signal during demodulation, we construct an orthogonal receiver based on frequency estimation to extract I/Q components, as follows:

$$x_I(nT_s) = x(nT_s)\cos(\omega_0 nT_s) = \sum_{z=1}^{Z_0} \frac{A_z}{2} d(nT_s - n_zT_s)[\cos(2\omega_0 nT_s + \varphi_z) + \cos(\varphi_z)] + n_{0,I}(nT_s) \tag{2}$$

$$x_Q(nT_s) = x(nT_s)\sin(\omega_0 nT_s) = \sum_{z=1}^{Z_0} \frac{A_z}{2} d(nT_s - n_zT_s)[\sin(2\omega_0 nT_s + \varphi_z) - \sin(\varphi_z)] + n_{0,Q}(nT_s) \tag{3}$$

Following the removal of high-frequency components, the signal can be expressed as follows:

$$X_I(nT_s) = \sum_{z=1}^{Z_0} \frac{A_z}{2} d(nT_s - n_zT_s)\cos(\varphi_z) + N_I(nT_s) \tag{4}$$

$$X_Q(nT_s) = -\sum_{z=1}^{Z_0} \frac{A_z}{2} d(nT_s - n_zT_s)\sin(\varphi_z) + N_Q(nT_s) \tag{5}$$

Although noise and multipath effects exist in underwater acoustic channels, they are independent of each other. However, at low SNRs, multipath effects are not the main influencing factor. Therefore, in this paper, the principles and advantages of the proposed method under the AWGN channel are presented in Section 2.2, and the multipath impact is briefly analyzed in Section 2.3.

2.2. *Average Second-Order Moment of Autocorrelation of Accumulated Code Difference Function*

2.2.1. Code Difference Function

1. Segmented Window

The i-th segmented window defined by a certain length is expressed as follows:

$$w(i, nT_s) = g(nT_s - iN_cT_s) = \begin{cases} 1 & iN_cT_s \le nT_s < (i+1)N_cT_s \\ 0 & others \end{cases} \tag{6}$$

Here, $g(nT_s)$ represents a rectangular pulse with a duration of N_cT_s.

The segmented window moves incrementally by one sampling interval (T_s). The schematic diagram of this process is illustrated in Figure 1.

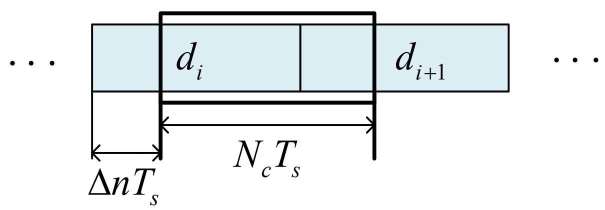


Figure 1. Schematic diagram of signal within i-th segmented window.

After $\Delta n (0 \le \Delta n < N_c)$ shifts, the I-channel baseband signal within the i-th segmented window can be expressed as follows:

$$x_w(i, \Delta n, nT_s) = x_I(nT_s)w(i, \Delta n, nT_s) = x_I(nT_s)g(nT_s - \Delta nT_s - iN_cT_s) \tag{7}$$

The i-th segmented window is divided into two parts (L_a and L_b), and the signals within it are also divided into two parts (front and back), as shown in Figure 2.

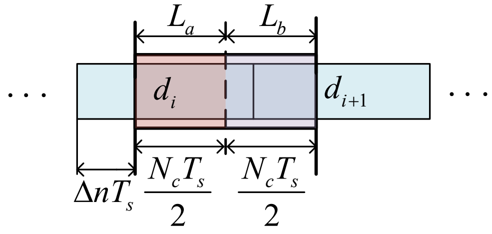


Figure 2. Schematic diagram of signal within i-th segmented window divided into two parts: front and back.

The front and back segmented window functions are expressed as follows:

$$w_{La}(i,\Delta n, nT_s) = g_a(nT_s - \Delta nT_s - iN_cT_s) = \begin{cases} 1 & \Delta nT_s + iN_cT_s \le nT_s < \Delta nT_s + (i+1/2)N_cT_s \\ 0 & others \end{cases} \tag{8}$$

$$w_{Lb}(i,\Delta n, nT_s) = g_a(nT_s - \Delta nT_s - N_cT_s/2 - iN_cT_s) = \begin{cases} 1 & \Delta nT_s + (i+1/2)N_cT_s \le nT_s < \Delta nT_s + (i+1)N_cT_s \\ 0 & others \end{cases} \tag{9}$$

Here, $g_a(nT_s)$ represents a rectangular pulse with a duration of $N_cT_s/2$.

The signals $X_{La}(i,\Delta n, nT_s)$ and $X_{Lb}(i,\Delta n, nT_s)$ that correspond to the L_a and L_b parts of the i-th segmented window are expressed as follows:

$$X_{La}(i,\Delta n, nT_s) = x_I(nT_s)w_{La}(i,\Delta n, nT_s) = x_I(nT_s)g_a(nT_s - \Delta nT_s - iN_cT_s) \tag{10}$$

$$X_{Lb}(i,\Delta n, nT_s) = x_I(nT_s)w_{Lb}(i,\Delta n, nT_s) = x_I(nT_s)g_a(nT_s - \Delta nT_s - N_cT_s/2 - iN_cT_s) \tag{11}$$

2. Code Difference Function

By summing the signals ($X_{La}(i,\Delta n, nT_s)$) of the L_a parts of the i-th segmented window, the corresponding sum functions ($S_{L_a}(i,\Delta n)$) can be obtained. Similarly, the $S_{L_b}(i,\Delta n)$ can also be obtained by summing $X_{Lb}(i,\Delta n, nT_s)$. The specific process is as follows:

- When $0 \le \Delta n < \frac{N_c}{2}$,

$$S_{L_a}(i,\Delta n) = \sum_{n=\Delta n}^{N_c/2-1+\Delta n} X_{La}(i,\Delta n, nT_s) = \frac{A}{2}\cos\varphi d_i \frac{N_c}{2} + N_{I,L_a}(i,\Delta n) \tag{12}$$

$$S_{L_b}(i,\Delta n) = \sum_{n=\Delta n+N_c/2}^{N_c-1+\Delta n} X_{Lb}(i,\Delta n, nT_s) = \frac{A}{2}\cos\varphi d_{i+1}\frac{\Delta n}{2} + \frac{A}{2}\cos\varphi d_i\left(\frac{N_c}{2} - \Delta n\right) + N_{I,L_b}(i,\Delta n) \tag{13}$$

- When $\frac{N_c}{2} \le \Delta n < N_c$,

$$S_{L_a}(i,\Delta n) = \sum_{n=\Delta n}^{N_c/2-1+\Delta n} X_{La}(i,\Delta n, nT_s) = \frac{A}{2}\cos\varphi d_i(N_c - \Delta n) + \frac{A}{2}\cos\varphi d_{i+1}\left(\Delta n - \frac{N_c}{2}\right) + N_{I,L_a}(i,\Delta n) \tag{14}$$

$$S_{L_b}(i,\Delta n) = \sum_{n=\Delta n+N_c/2}^{N_c-1+\Delta n} X_{Lb}(i,\Delta n, nT_s) = \frac{A}{2}\cos\varphi d_{i+1}\frac{N_c}{2} + N_{I,L_b}(i,\Delta n) \tag{15}$$

By calculating the difference between $S_{L_a}(i,\Delta n)$ and $S_{L_b}(i,\Delta n)$ of the i-th segmented window, the code difference function (CDF) can be expressed as follows:

$$\Delta S_I(i,\Delta n) = S_{L_a}(i,\Delta n) - S_{L_b}(i,\Delta n) = \frac{A}{2}\cos\varphi(d_i - d_{i+1})\begin{cases} \Delta n & 0 \le \Delta n < N_c/2 \\ N_c - \Delta n & N_c/2 \le \Delta n < N_c \end{cases} + N_I'(i,\Delta n) \tag{16}$$

Simulation verification: The DSSS/BPSK signal was characterized by a 10 kHz carrier frequency, a 0.5 ms chip width, an 80 kHz sampling rate, a spreading spectrum code of a fifth-order m-sequence, and a total of five source information codes, and the background noise was modeled as Gaussian white noise with a signal-to-noise ratio of 10 dB. The code difference results corresponding to the different sliding times (Δn) in the sliding process of the i-th segmented window are shown in Figure 3.

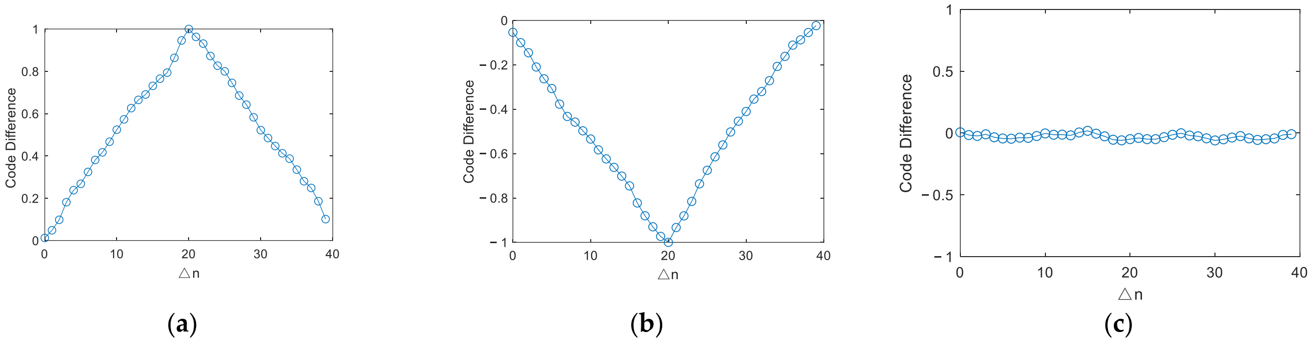


Figure 3. Variations in code differences within i-th segmented window with sliding times: (**a**) $d_i = 1,\ d_{i+1} = -1$; (**b**) $d_i = -1,\ d_{i+1} = 1$; (**c**) $d_i = d_{i+1}$.

As shown above, with the increase in the Δn, the variation in the code difference value of the i-th segmented window is related to d_i and d_{i+1}. In Figure 3a, the code difference value first increases and then decreases, reaching its maximum value at $\Delta n = T_c/2$. In Figure 3b, the code difference value first decreases and then increases, reaching its minimum value also at $\Delta n = T_c/2$. In Figure 3c, there is no extremum in the code difference function, only fluctuations caused by noise, which is consistent with the theoretical analysis of Equation (16).

2.2.2. Accumulation of Code Difference Functions

According to the expression of the code difference function and the above analysis, although the peak heights are different, when adjacent chips change, the code difference function corresponding to different sliding times (Δn) always exhibits peaks. The code difference functions demonstrate strong correlation. However, the correlation between the code difference functions of noise is weak. Therefore, the peaks of the code difference function become more distinct through accumulation.

In this section, we analyze the power of the accumulated code difference functions of the signal and noise.

The accumulation results of the code difference functions for the I and Q channels are expressed as follows:

$$S_I(i) = \sum_{\Delta n=0}^{N_c-1} S_I(i,\Delta n) = A\cos\varphi(d_i - d_{i+1})N_c^2/8 + N_{sI}(i) \tag{17}$$

$$S_Q(i) = \sum_{\Delta n=0}^{N_c-1} S_Q(i,\Delta n) = -A\sin\varphi(d_i - d_{i+1})N_c^2/8 + N_{sQ}(i) \tag{18}$$

According to the above equations, the peaks of the code difference functions of the DSSS signals of the I and Q channels are $\frac{A\cos\varphi N_c^2}{4}$ and $\frac{A\sin\varphi N_c^2}{4}$ at the time of the code change, respectively.

Because the segmented windows corresponding to distinct sliding distances encompass overlapping noise segments, the code difference functions of noise for these various sliding distances are not statistically independent. Consequently, the variance of the accumulated code difference function for noise is expressed as follows:

$$Var(N_{sI}) = \sum_{\Delta n=0}^{N_c-1} Var\left(N_I'(\Delta n)\right) + 2\sum_{0\le\Delta n_p<\Delta n_q\le N_c-1} Cov\left(N_I'(\Delta n_p), N_I'(\Delta n_q)\right) \quad (19)$$

According to the independence of Gaussian white noise, under a sliding distance of Δn, $N_I'(\Delta n, i)$ follows a Gaussian distribution with a mean of zero and a variance of $N_c\sigma_n^2/2$. Therefore, the above expression can be represented as follows:

$$Var(N_{sI}) = N_c N_c \sigma_n^2/2 + 2\sum_{0\le\Delta n_p<\Delta n_q\le N_c-1} E\left(N_I'(\Delta n_p, i)N_I'(\Delta n_q, i)\right) \quad (20)$$

where

$$Cov\left(N_I'(\Delta n_p, i), N_I'(\Delta n_q, i)\right) = E\left(N_I'(\Delta n_p, i)N_I'(\Delta n_q, i)\right) = E\left[\left(N_{I,L_a}(\Delta n_p, i) - N_{I,L_b}(\Delta n_p, i)\right)\left(N_{I,L_a}(\Delta n_q, i) - N_{I,L_b}(\Delta n_q, i)\right)\right] \quad (21)$$

Since the noise at each moment is an independent and identically distributed random variable, only the overlapping segments of the two sequences corresponding to different sliding distances $(\Delta n_p$ and $\Delta n_q)$ contribute to the covariance. The length of the overlapping segments, along with the sum of the difference in the sliding distances between the two sequences, equals the chip width (N_c). Therefore, by defining the difference ($d = \Delta n_q - \Delta n_p$) in the sliding distances, the covariance between $N_I'(\Delta n_p, i)$ and $N_I'(\Delta n_q, i)$ can be expressed as follows:

- When $d \le N_c/2$,

$$\begin{aligned} &Cov\left(N_I'(\Delta n_p, i), N_I'(\Delta n_q, i)\right) \\ &= E\left(N_{I,La}(\Delta n_p, i)N_{I,La}(\Delta n_p + d, i)\right) + E\left(-N_{I,Lb}(\Delta n_p, i)N_{I,La}(\Delta n_p + d, i)\right) + E\left(N_{I,Lb}(\Delta n_p, i)N_{I,Lb}(\Delta n_p + d, i)\right) \\ &= (N_c - 3d)E\left((N_I(nT_s))^2\right) \end{aligned} \quad (22)$$

- When $d > N_c/2$,

$$Cov\left(N_I'(\Delta n_p, i), N_I'(\Delta n_q, i)\right) = E\left(-N_{I,Lb}(\Delta n_p, i)N_{I,La}(\Delta n_p + d, i)\right) = -(N_c - d)E\left((N_I(nT_s))^2\right) \quad (23)$$

By substituting Equations (22) and (23) into Equation (19), when the noise power of the I channel is $\sigma_n^2/2$, the power of its accumulated code difference function can be expressed as follows:

$$\begin{aligned} Var(N_{sI}) &= \sum_{\Delta n=0}^{N_c-1} Var(N_I'(\Delta n)) + 2\sum_{d=1}^{N_c-1}(N_c - d)Cov\left(N_I'(\Delta n_p, i), N_I'(\Delta n_q, i)\right) \\ &= N_c^2\sigma_n^2/2 + \sigma_n^2\sum_{d=1}^{N_c/2}(N_c - d)(N_c - 3d) - \sigma_n^2\sum_{d=N_c/2+1}^{N_c-1}(N_c - d)(N_c - d) \\ &= \sigma_n^2/2\left(\frac{N_c^3}{6} + \frac{N_c}{3}\right) \end{aligned} \quad (24)$$

Likewise, the power of the accumulated code difference function of the Q-channel noise can be expressed as follows:

$$Var(N_{sQ}) = \sigma_n^2/2\left(\frac{N_c^3}{6} + \frac{N_c}{3}\right) \quad (25)$$

The m-sequence is a common spread spectrum sequence. The length of an n-order m-sequence is $L = 2^n - 1$. For a signal with one PN sequence period length, the accumulated code difference function sequence (S_I') is of the length $L-1$, which can be expressed as follows:

$$S_I' = \{S_I(1), \cdots, S_I(i), \cdots, S_I(L-1)\}_{1\times L-1} \tag{26}$$

According to its run-length distribution properties, the number of code changes in an n-order m-sequence is $2^{n-1}-1$. Therefore, the signal-to-noise ratio in S_I' can be expressed as follows:

$$\gamma_{ACDF} = \frac{P_{Isig}+P_{Qsig}}{P_{Inoise}+P_{Qnoise}} = \frac{\left(\frac{AN_c^2}{4}\right)^2/2}{\sigma_n^2\left(\frac{N_c^3}{6}+\frac{N_c}{3}\right)} = \frac{A^2}{\sigma_n^2}\frac{3N_c^3/2}{8(N_c^2+2N_c)} \tag{27}$$

where A^2/σ_n^2 is the signal-to-noise ratio of the input baseband signal.

Then, we validated the accuracy of the above formula through simulation. The simulation parameters remained the same as those described above. A comparison between the theoretical and simulated results of the γ_{ACDF} under different SNRs is presented in Figure 4.

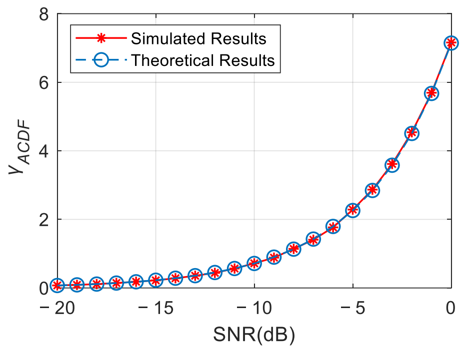


Figure 4. Comparison of theoretical and simulated results of γ_{ACDF}.

According to Figure 4, the simulated results are consistent with the theoretical results (i.e., Equation (27)), indicating that the γ_{ACDF} can accurately represent the signal-to-noise ratio of the ACDF. Moreover, according to the expression of the γ_{ACDF}, the signal-to-noise ratio of the ACDF is significantly improved compared with that of the input baseband signal in actual underwater acoustic reconnaissance scenarios (with high sampling rates and sufficient N_c).

2.2.3. Autocorrelation Characteristics of Accumulated Code Difference Function

When the observed signal spans $j_0(j_0 \geq 2)$ PN sequence periods, the accumulated code difference function can be expressed as follows:

$$S_I(i) = A\cos\varphi(d_i - d_{i+1})N_c^2/8 + N_{sI}(i)\ i = 1,2,3,\ldots,L,\ldots,2L,\ldots,j_0L-1 \tag{28}$$

The autocorrelation function of $S_I(i)$ can be obtained as follows:

$$R_I(k) = \sum_{i=1}^{j_0L-1} S_I(i)S_I(i+k) = \left(\frac{N_c^2}{8}\cos\varphi\right)^2[2R_d(k) - R_d(k+1) - R_d(k-1)] + R_{NI}(k) \tag{29}$$

Similarly,

$$R_Q(k) = \left(-\frac{N_c^2}{8}\sin\varphi\right)^2[2R_d(k) - R_d(k+1) - R_d(k-1)] + R_{NQ}(k) \tag{30}$$

The autocorrelation results of m-sequences can be expressed as follows:

$$R_d(k) = \begin{cases} 1 & k = jL \\ -1/L & k \neq jL \end{cases} \tag{31}$$

Here, $L = 2^n - 1$ represents the period of an n-order PN sequence, and $j = 1, \ldots, j_0$ is an integer.

Using the m-sequence as an example, the normalized result of summing the autocorrelation functions of the I/Q channels can be expressed as follows:

$$R(k) = R_I(k) + R_Q(k) = [2R_d(k) - R_d(k+1) - R_d(k-1)] + R_N(k) = R_N(k) + \begin{cases} 2 + 2/L & k = jL \\ -1 - 1/L & k = jL \pm 1 \\ 0 & else \end{cases} \tag{32}$$

According to the above formulation, the $R(k)$ exhibits correlation peaks at intervals corresponding to the PN sequence period when the observed signal spans multiple PN sequence periods.

The autocorrelation of the ACDF is illustrated in Figure 5, employing the same simulation parameters as those described above.

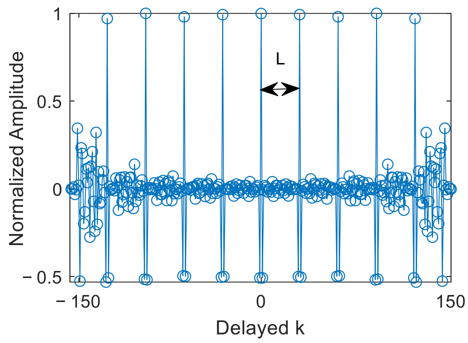


Figure 5. Autocorrelation characteristics of the accumulated code difference function.

As shown in Figure 5, the ACDF autocorrelation exhibits correlation peaks, and the intervals between adjacent peaks correspond to the PN sequence period (L = 31), which is consistent with the theoretical predictions (i.e., Equation (32)).

To further analyze the advantage of the autocorrelation peaks of the ACDF, we represent the degree of separation between the peaks and the baseline fluctuations via the ratio of the peak amplitude to the baseline standard deviation, known as the peak-to-base ratio (PBR) [25]. The simulation parameters are the same as those described above, and a comparison of the PBRs of the autocorrelation of the ACDF and that of the original baseband signal (i.e., the existing autocorrelation method) is presented in Figure 6.

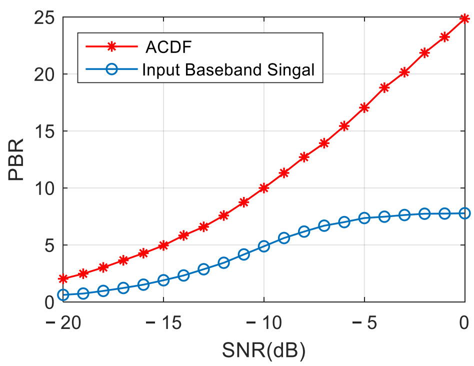


Figure 6. PBRs of ACDF autocorrelation and input baseband signal autocorrelation vary with SNR.

As shown in Figure 6, under the same SNR, the ACDF autocorrelation has a better PBR than the input baseband signal autocorrelation. The higher the PBR, the better the peak performance, which is beneficial for achieving peak extraction. Therefore, the autocorrelation results of the ACDF are more conducive to achieving PN sequence period estimation.

2.2.4. Average Second-Order Moment of ACDF Autocorrelation

In practical applications, the randomness of the source code sequence can cause positive and negative peaks to cancel each other out at integer multiples of the PN sequence period. Additionally, the signal's autocorrelation function is estimated using limited sample data, due to the finite observation time, which may lead to suboptimal noise suppression [23].

To address the issues mentioned above, in our method, the average second-order moment of the autocorrelation of the accumulated code difference function is constructed as $\rho(k)$, which is expressed as follows:

$$\rho(k) = \frac{1}{M}\sum_{m=1}^{M}[R^m(k)]^2 \tag{33}$$

where m represents the autocorrelation function of the accumulated code difference for the m-th segment of the received signal, and M denotes the number of accumulations in the ensemble averaging method. As analyzed above, $\rho(k)$ contains correlation peaks at intervals corresponding to the PN sequence period, and the period can be estimated by extracting these intervals.

Simulation verification: The DSSS/BPSK signal was characterized by a 0.5 ms chip width, a 80 kHz sampling rate, a spreading spectrum code based on a fifth-order m-sequence, an SNR of -10 dB, and 20 randomly generated source information codes. The results of the ASOMA-ACDF when $M = 1$ and $M = 5$ are shown in Figures 7 and 8.

A comparison of Figures 7 and 8 reveals that autocorrelation peaks, spaced by L (i.e., the PN sequence period), are evident in both methods, consistent with the theoretical predictions. However, the presence of noise and the inherent randomness of the information code obscure certain correlation peaks in Figure 7, diminishing their clarity to the point that they cannot be extracted, which is not conducive to estimating the peak interval that is equal to the PN sequence period. In contrast, the autocorrelation peaks in Figure 8 are more distinct. Therefore, the ASOMA-ACDF achieves high-precision PN sequence period estimation.

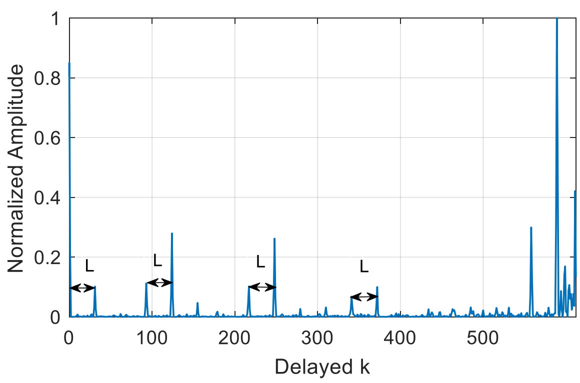


Figure 7. Average second-order moment of autocorrelation of accumulated code difference function $(M = 1)$.

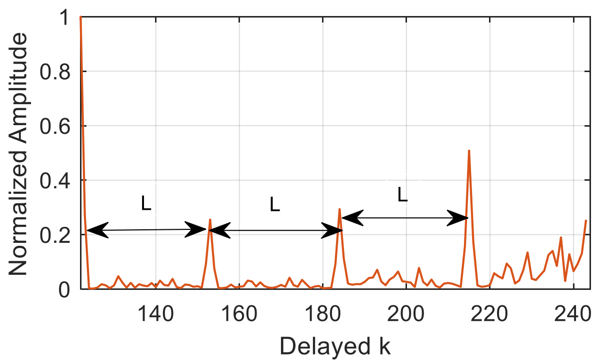


Figure 8. Average second-order moment of autocorrelation of accumulated code difference function ($M = 5$).

2.3. Multipath Effect on ACDF Autocorrelation Characteristics

The time delay of the z-th path signal is $n_z T_s$. However, the starting position of the segmented window is consistent with the starting time of the first path signal. Therefore, there is an asynchronous time ($n_z T_s$) between the z-th path signal and the segmented window. Due to the time delay, the i-th window sliding process involves three chips, which is different from the two chips mentioned in Section 2.2. Therefore, the time delay of multipath signals is the fundamental issue that affects the ACDF and its autocorrelation characteristics. We briefly analyze the impact of this time delay on the ACDF under the z-th path below. Because the derivation process is similar to that described in Section 2.2, we only briefly present the results in this section.

The z-th path signals $X_{La}(i, \Delta n, nT_s, z)$ and $X_{Lb}(i, \Delta n, nT_s, z)$, corresponding to the L_a and L_b parts in the i-th segmented window, respectively, are expressed as follows:

$$X_{La}(i, \Delta n, nT_s, z) = \frac{A_z}{2} d(nT_s - n_z T_s) \cos(\varphi_z) w_{La}(i, \Delta n, nT_s) \tag{34}$$

$$X_{Lb}(i, \Delta n, nT_s, z) = \frac{A_z}{2} d(nT_s - n_z T_s) \cos(\varphi_z) w_{Lb}(i, \Delta n, nT_s) \tag{35}$$

Assuming that the delay (n_z) of the z-th path can be represented as $n_z = K_z N_c + \tau_z$, where $K_z \in \mathbb{Z}$, $0 \le \tau_z < N_c$, the accumulated code difference function of the z-th path can be expressed as follows:

- When $0 \le \tau_z < N_c/2$,

$$\begin{aligned} S_I(i,z) &= \sum_{\Delta n=0}^{N_c-1} \Delta S(i,\Delta n,z) = \sum_{\Delta n=0}^{N_c-1} [S_{La}(i,\Delta n,z) - S_{Lb}(i,\Delta n,z)] = \sum_{\Delta n=0}^{N_c-1} \left(\sum_{n=0}^{N_c/2-1} X_{La}(i,\Delta n,nT_s,z) - \sum_{N_c/2}^{N_c-1} X_{Lb}(i,\Delta n,nT_s,z) \right) \\ &= \frac{A_z}{2} \cos\varphi_z \left\{ \frac{\tau_z(\tau_z+1)}{2} d_{i-K_z-1} + [N_c^2/4 - \tau_z(\tau_z+1)] d_{i-K_z} + \left[\frac{\tau_z(\tau_z+1)}{2} - N_c^2/4 \right] d_{i-K_z+1} \right\} \end{aligned} \tag{36}$$

- When $N_c/2 \le \tau_z < N_c$,

$$S_I(i,z) = \frac{A_z}{2} \cos\varphi_z \left\{ \left[N_c^2/4 - \frac{(N_c-\tau_z)(N_c-\tau_z-1)}{2} \right] d_{i-K_z-1} - \frac{(N_c-\tau_z)(N_c-\tau_z-1)}{2} d_{i-K_z+1} - \left[N_c^2/4 - (N_c-\tau_z)(N_c-\tau_z-1) \right] d_{i-K_z} \right\} \tag{37}$$

As can be seen from Equations (36) and (37), due to the time delay influence, the CDF of the z-th path signal cannot achieve the ideal accumulation, as it did in Section 2.2, resulting in a loss of ACDF gain. However, the first path ($n_z = 0$) can still achieve ideal accumulation (as shown in Section 2.2).

In multipath channels, the received signal is the result of the linear addition of the signals from each path, and the calculation of the ACDF is also a linear process. Therefore, by linearly adding the ACDF of each path, the ACDF of the received multipath signal can be obtained, which is expressed as follows:

$$S_{ZI}(i) = \sum_{z=1}^{Z_0} S_I(i,z) = \sum_{z=1}^{Z_0} \frac{A_z}{2} \cos\varphi_z \left(W_{\tau_z}^{i-K_z-1} d_{i-K_z-1} - W_{\tau_z}^{i-K_z} d_{i-K_z} + W_{\tau_z}^{i-K_z+1} d_{i-K_z+1} \right) \quad (38)$$

where the weight of d_{i-K_z-1} is $W_{\tau_z}^{i-K_z-1}$, the weight of d_{i-K_z} is $W_{\tau_z}^{i-K_z}$, and the weight of d_{i-K_z+1} is $W_{\tau_z}^{i-K_z+1}$. The three weights of the z-th path can be obtained from Equations (36) and (37) based on the time delay (n_z).

In multipath channels, the ACDF autocorrelation of the I-channel signal can be expressed as follows:

$$R_I(k) = \sum_{i=-\infty}^{\infty} \left[\sum_{z_m=1}^{Z_0} S_I(i,z_m) \sum_{z_n=1}^{Z_0} S_I(i+k,z_n) \right] = \sum_{z_m=1}^{Z_0} \sum_{z_n=1}^{Z_0} \left(\sum_{i=-\infty}^{\infty} S_I(i,z_m) S_I(i+k,z_n) \right) = \sum_{z_m=1}^{Z_0} \sum_{z_n=1}^{Z_0} R_{I-z_m z_n}(k) \quad (39)$$

The ACDF autocorrelation of the Q-channel signal can be similarly obtained. The specific expression for the summation result of the autocorrelation functions of the I/Q channels is as follows:

$$\begin{aligned} R(k) &= \sum_{z_m=1}^{Z_0} \sum_{z_n=1}^{Z_0} \left(R_{I-z_m z_n}(k) + R_{Q-z_m z_n}(k) \right) \\ &= \sum_{z_m=1}^{Z_0} \sum_{z_n=1}^{Z_0} \frac{A_{z_m} A_{z_n} \cos(\varphi_{z_m} - \varphi_{z_n})}{4} \begin{bmatrix} r_{K_{z_m}-K_{z_n}} R_d(K_{z_m} - K_{z_n} + k) + r_{K_{z_m}-K_{z_n}-1} R_d(K_{z_m} - K_{z_n} + k - 1) \\ + r_{K_{z_m}-K_{z_n}+1} R_d(K_{z_m} - K_{z_n} + k + 1) + r_{K_{z_m}-K_{z_n}-2} R_d(K_{z_m} - K_{z_n} + k - 2) \\ + r_{K_{z_m}-K_{z_n}+2} R_d(K_{z_m} - K_{z_n} + k + 2) \end{bmatrix} \end{aligned} \quad (40)$$

where

$$\begin{aligned} r_{K_{z_m}-K_{z_n}} &= W_{\tau_{z_m}}^{i-K_{z_m}} W_{\tau_{z_n}}^{i-K_{z_n}} + W_{\tau_{z_m}}^{i-K_{z_m}-1} W_{\tau_{z_n}}^{i-K_{z_n}-1} + W_{\tau_{z_m}}^{i-K_{z_m}+1} W_{\tau_{z_n}}^{i-K_{z_n}+1} \\ r_{K_{z_m}-K_{z_n}-1} &= W_{\tau_{z_m}}^{i-K_{z_m}} W_{\tau_{z_n}}^{i-K_{z_n}-1} + W_{\tau_{z_m}}^{i-K_{z_m}+1} W_{\tau_{z_n}}^{i-K_{z_n}} \\ r_{K_{z_m}-K_{z_n}+1} &= W_{\tau_{z_m}}^{i-K_{z_m}} W_{\tau_{z_n}}^{i-K_{z_n}+1} + W_{\tau_{z_m}}^{i-K_{z_m}-1} W_{\tau_{z_n}}^{i-K_{z_n}} \\ r_{K_{z_m}-K_{z_n}-2} &= W_{\tau_{z_m}}^{i-K_{z_m}+1} W_{\tau_{z_n}}^{i-K_{z_n}-1} \\ r_{K_{z_m}-K_{z_n}+2} &= W_{\tau_{z_m}}^{i-K_{z_m}-1} W_{\tau_{z_n}}^{i-K_{z_n}+1} \end{aligned}$$

As can be seen from Equation (40), the autocorrelation result of the ACDF in multipath channels is the sum of the autocorrelation results of the ACDF in each path $(z_m = z_n)$ and the cross-correlation results of any two paths $(z_m \neq z_n)$. Based on the characteristics of $R_d(k)$ (i.e., Equation (31)), it can be inferred that the autocorrelation results of each path all have peaks at $k = jL$, thus enabling peak accumulation. Although the interval between the cross-correlation peaks of two paths (z_m and z_n) is also L, the peak position depends on the delay difference ($K_{z_m} - K_{z_n}$), thus enabling peak accumulation. Therefore, these cross-correlated peaks are often lower than the main peak and often appear as fake peaks in the intervals between the adjacent main peaks.

Simulation verification: Assume multipath channels with the following parameters:

Channel 1: total number of paths: three; corresponding amplitude: $A_z = [1,\ 0.9,\ 1.1]$; corresponding delay: $n_z = [K_1 + \tau_1,\ K_2 + \tau_2,\ K_3 + \tau_3] = [0,\ 14N_c + 4,\ 34N_c + 14]$.

Channel 2: total number of paths: three; corresponding amplitude: $A_z = [1,\ 0.5,\ 0.25]$; corresponding delay: $n_z = [K_1 + \tau_1,\ K_2 + \tau_2,\ K_3 + \tau_3] = [0,\ 6N_c + 8,\ 21N_c + 3]$.

Employing the same simulation DSSS signal parameters as those in Section 2.2, the ACDF autocorrelation in the two multipath channels are illustrated in Figure 9.

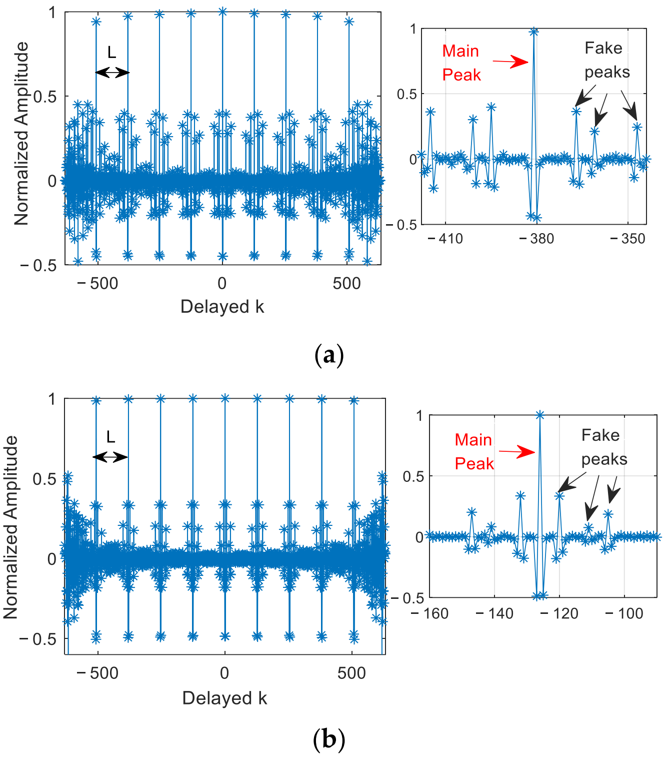


Figure 9. ACDF autocorrelation under multipath channel. (**a**) Channel 1. (**b**) Channel 2.

Figure 9 shows that there are still significant main peaks in the autocorrelation results of the ACDF under multipath channels, with PN sequence periods as intervals. Therefore, the proposed method can still achieve PN sequence period estimation by extracting peak positions in multipath channels. Between the adjacent main peaks, there are some lower fake peaks, which are caused by the cross-correlation between the ACDFs of different paths. According to the theoretical analysis (Equation (40)), the intervals between the fake and main peaks should be $K_2 - K_1, K_3 - K_2$, and $K_3 - K_1$. A specific analysis of Figure 9a reveals that the intervals between the three fake-peaks and their adjacent main peak are 14, 20, and 34 in Channel 1, respectively. A specific analysis of Figure 9b reveals that hat the intervals between the three fake-peaks and their adjacent main peak are 6, 15, and 21 in Channel 2, respectively. From this, it can be seen that the positions of peaks are consistent with the theoretical analysis.

When the amplitudes of multipath channel are similar (as shown in Figure 9a), the amplitudes of the fake-peaks are lower than that of the main peak, and the position of the main peak can be extracted. When the amplitudes of multipath channel are different (as shown in Figure 9b), the amplitudes of the fake-peaks are still lower than that of the main peak, and the height of the fake-peak generated by cross-correlation between multipath signals with lower amplitudes is also lower, making it easier to extract the position of the main peak. Therefore, it can be seen that since the main peak is the result of the accumulation of autocorrelation peaks of each path signal, the amplitude of the fake-peaks generated by the cross-correlation of any two paths are always relatively lower than that of the main peak.

3. Implementation Process

Based on the above theoretical analysis, we propose a method utilizing the average second-order moment of the autocorrelation of the accumulated code difference function (ASOMA-ACDF). The detailed implementation process is outlined in the flowchart in Figure 10.

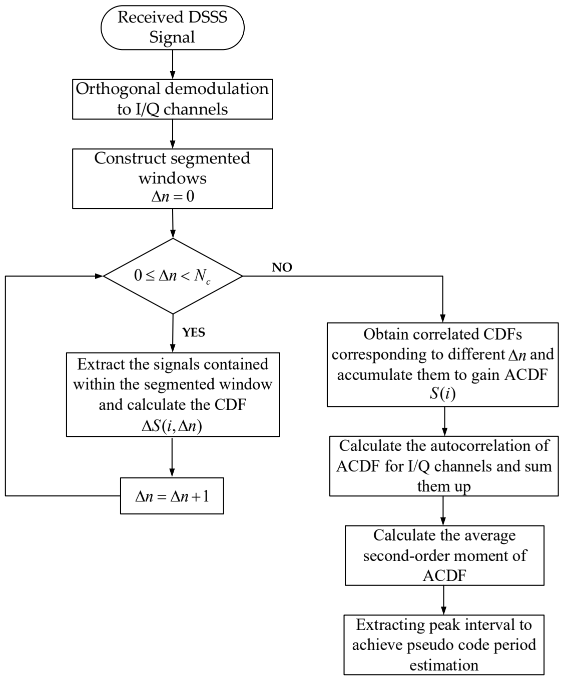


Figure 10. Flowchart of ASOMA-ACDF process.

4. Simulation Analysis

Simulation parameters: The signal emitted by the transmitter was a DSSS/BPSK signal. The chip rate of the baseband signal was 2 kHz, and the carrier frequency for modulation was 10 kHz. The bandwidth of the non-cooperative receiver was 4–16 kHz, the sampling rate was 70 kHz, and the number of source information symbols was 90. The background noise was Gaussian white noise, and the number of Monte Carlo simulations was 2000.

The root-mean-square error (RMSE) and correct estimation probability were used to compare the performances of the average second-order-moment autocorrelation (ASOMA) [16,17,26], accumulated-power-spectrum-reprocessing (A-PSR) [18,19], and average second-order-moment autocorrelation of accumulated code difference function (ASOMA-ACDF) methods. The performance metrics are defined as follows:

The correct probability of the PN sequence period estimation is as follows:

$$P_e = P\left(|\hat{T}_p - T_p| \le T_c\right) \tag{41}$$

The root-mean-square (RMS) error of the estimation of the PN sequence period is as follows:

$$RMSE = \sqrt{\frac{1}{N_k}\sum_{k=1}^{N_k}\left(\hat{T}_p - T_p\right)^2} \tag{42}$$

where T_c is the chip width, and T_p and $\hat{T}_p$ represent the true and estimated values of the PN sequence period, respectively.

4.1. Effects of Different PN Sequence Lengths on Performance

In this experiment, the source information codes were randomly generated. To reduce the impact of the inherent randomness of the source information code on the PN sequence period estimation accuracy, the ensemble averaging method, with $M = 5$ accumulations, was applied to the three methods. The estimation performances of the three methods under PN sequences of different lengths are presented below.

1. For a fifth-order m-sequence.

As shown in Figures 11 and 12, when a fifth-order m-sequence served as the spreading sequence, the estimation performances of all three methods improved with the increase in the SNR. Among them, the ASOMA-ACDF achieved the best performance, followed by the ASOMA, while the A-PSR was slightly inferior to the ASOMA. Specifically, when the SNR was −11 dB (Figure 11), the correct estimation probability of the ASOMA-ACDF was 0.928, compared with 0.662 for the ASOMA and 0.566 for the A-PSR. The root-mean-square error (RMSE) analysis (Figure 12) reveals corresponding values of 0.165 ms for the ASOMA-ACDF and approximately 0.276 ms for both the ASOMA and A-PSR at the same SNR level. As the SNR continued to increase, the root-mean-square error of the ASOMA gradually became better than that of the A-PSR, though both remained inferior to that of the ASOMA-ACDF. When the SNR reached −8 dB, the root-mean-square error of the ASOMA-ACDF decreased to 0.04 ms, while that of the ASOMA decreased to 0.078 ms, and that of the A-PSR was approximately 0.1 ms.

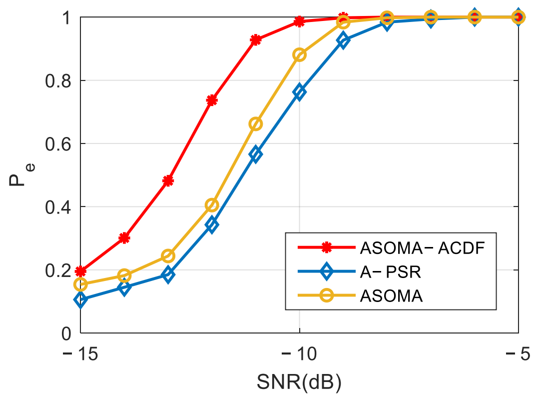


Figure 11. Variation in correct estimation probabilities with SNR for a fifth-order m-sequence.

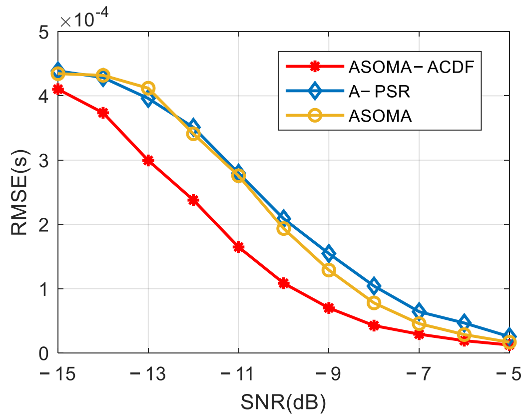


Figure 12. Variation in root-mean-square errors with SNR for a fifth-order m-sequence.

2. For a seventh-order m-sequence:

As shown in Figures 13 and 14, when a seventh-order m-sequence served as the spreading sequence, the estimation performances of all three methods improved with the increase in the SNR. Among them, the ASOMA-ACDF achieved the best performance, followed by the ASOMA, while the A-PSR was slightly inferior to the ASOMA. Specifically, when the SNR was −14 dB (Figure 13), the correct estimation probability of the ASOMA-ACDF was 0.916, compared with 0.638 for the ASOMA and 0.46 for the A-PSR. The root-mean-square error (RMSE) analysis (Figure 14) reveals corresponding values of

0.76 ms for the ASOMA-ACDF and approximately 1.4 ms for both the ASOMA and A-PSR when the SNR was −16 dB. As the SNR continued to increase, the root-mean-square error of the ASOMA gradually became better than that of the A-PSR, though both remained inferior to the ASOMA-ACDF. When the SNR reached −11 dB, the root-mean-square error of the ASOMA-ACDF decreased to 0.015 ms, while that of the ASOMA decreased to 0.032 ms, and that of the A-PSR was approximately 0.057 ms.

3. For a ninth-order m-sequence:

As shown in Figures 15 and 16, when a ninth-order m-sequence served as the spreading sequence, the estimation performances of all three methods improved with the increase in the SNR. Among them, the ASOMA-ACDF achieved the best performance, followed by the ASOMA, while the A-PSR was slightly inferior to the ASOMA. Specifically, when the SNR was −17 dB (Figure 15), the correct estimation probability of the ASOMA-ACDF was 0.964, compared with 0.532 for the ASOMA and 0.346 for the A-PSR. The root-mean-square error (RMSE) analysis (Figure 16) reveals corresponding values of 1.4 ms for the ASOMA-ACDF and approximately 4 ms for both the ASOMA and A-PSR when the SNR was −18 dB. As the SNR continued to increase, the root-mean-square error of the ASOMA gradually became better than that of the A-PSR, though both remained inferior to that of the ASOMA-ACDF. When the SNR reached −14 dB, the root-mean-square error of the ASOMA-ACDF decreased to 0.005 ms, while that of the ASOMA decreased to 0.024 ms, and that of the A-PSR was approximately 0.042 ms.

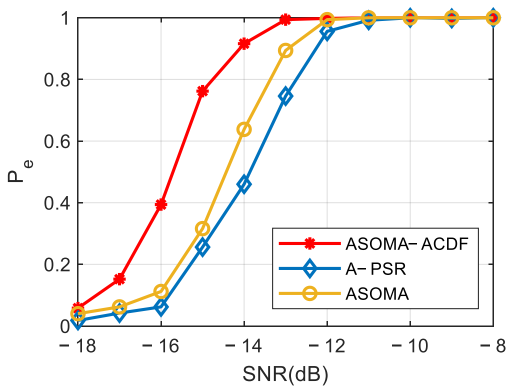


Figure 13. Variation in correct estimation probabilities with SNR for a seventh-order m-sequence.

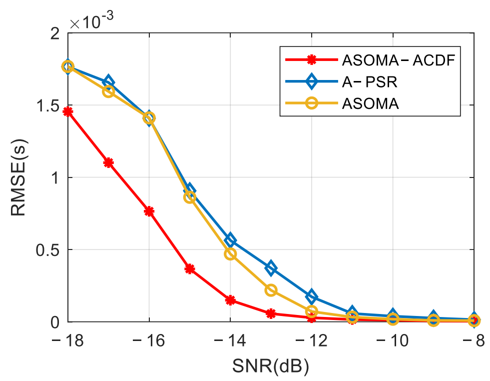


Figure 14. Variation in root-mean-square errors with SNR for a seventh-order m-sequence.

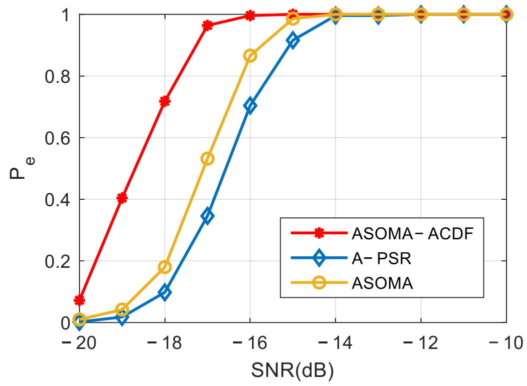


Figure 15. Variation in correct estimation probabilities with SNR for a ninth-order m-sequence.

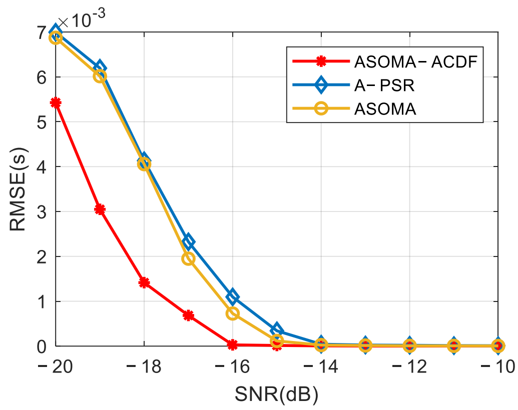


Figure 16. Variation in root-mean-square errors with SNR for a ninth-order m-sequence.

4.2. Effects of Different Accumulation Counts on Estimation Performance

Figure 17 shows that the correct estimation probabilities for all three methods improved as the accumulation count increased. Among them, the ASOMA-ACDF maintained the best performance, followed by the ASOMA, while the A-PSR was slightly inferior to the ASOMA. Specifically, the correct estimation probabilities of the three methods were as follows when the SNR was −14 dB: ASOMA-ACDF: 0.524, ASOMA: 0.224, and A-PSR: 0.138 for an accumulation count of 1; ASOMA-ACDF: 0.916, ASOMA: 0.638, and A-PSR: 0.46 for an accumulation count of 5; ASOMA-ACDF: 0.978, ASOMA: 0.8, and A-PSR: 0.702 for an accumulation count of 10.

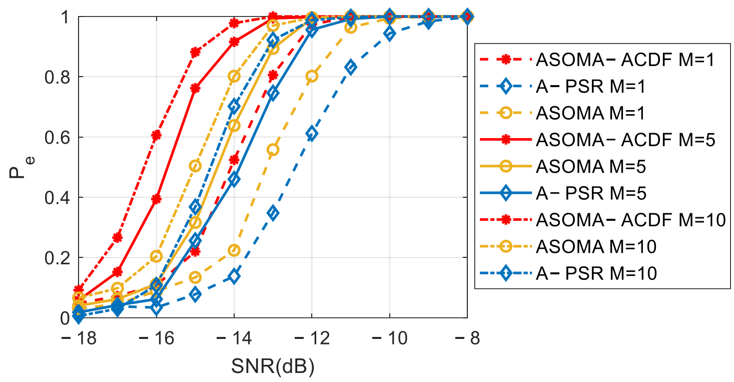


Figure 17. Variation in correct estimation probabilities with SNR at different accumulation counts.

Figure 18 shows that the root-mean-square (RMS) errors for all three methods decreased as the accumulation count increased. Among them, the ASOMA-ACDF maintained the best performance, followed by the ASOMA, while the A-PSR was slightly inferior to the ASOMA. Specifically, the corresponding root-mean-square (RMS) errors for the three methods were as follows when the SNR was −14 dB: ASOMA-ACDF: 0.61 ms, ASOMA: approximately 1.1 ms, and A-PSR: 1.2 ms for an accumulation count of 1; ASOMA-ACDF: 0.15 ms, ASOMA: 0.47 ms, and A-PSR: approximately 0.56 ms for an accumulation count of 5; ASOMA-ACDF: 0.048 ms, ASOMA: 0.3 ms, and A-PSR: approximately 0.38 ms for an accumulation count of 10.

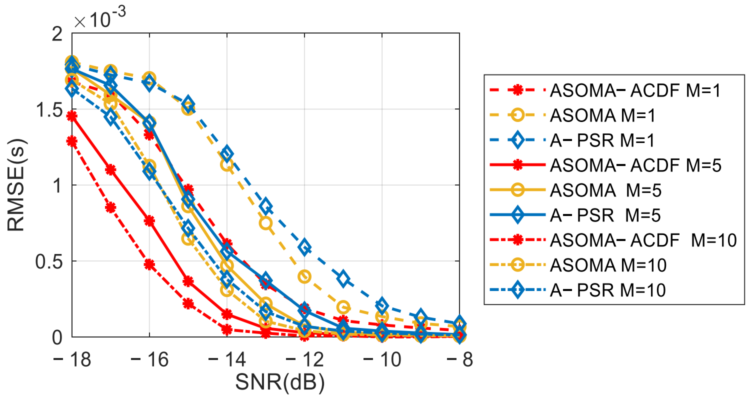


Figure 18. Variation in root-mean-square errors with SNR at different accumulation counts.

4.3. Effect of Frequency Estimation Errors on Estimation Performance

In this experiment, the frequency estimation errors were uniformly distributed in the range of −500 Hz~500 Hz (when carrier frequency is 10 kHz). The accumulation count was 5, and the remaining parameters were the same as those described above. The performances of the three methods when the PN sequence was a seventh-order m-sequence are presented in Figure 19.

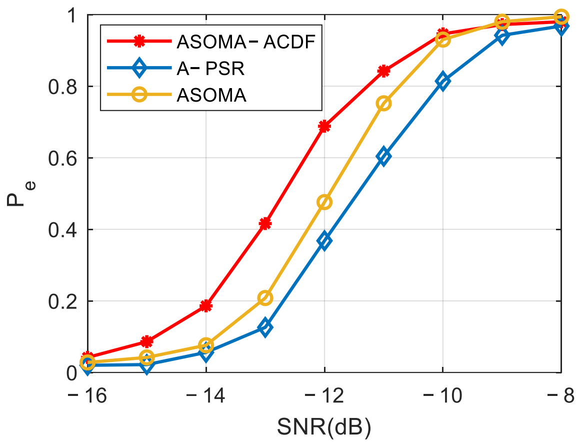


Figure 19. Variation in correct estimation probabilities with SNR when frequency was imperfectly estimated.

As shown in Figures 19 and 20, when the frequency was imperfectly estimated, the performances of all three methods improved with the increase in the SNR. Among them, the ASOMA-ACDF achieved the best performance, followed by the ASOMA, while the A-PSR was slightly inferior to the ASOMA. Specifically, when the SNR was −11 dB (Figure 19), the correct estimation probability of the ASOMA-ACDF was 0.842, compared with 0.752 for the ASOMA and 0.604 for the A-PSR. The root-mean-square error (RMSE) analysis (Figure 20) reveals corresponding values of 0.635 ms for the ASOMA-ACDF and approximately 1.09 ms for both the ASOMA and A-PSR when the SNR was −13 dB. As the SNR continued to increase, the root-mean-square error of the ASOMA gradually became better than that of the A-PSR, though both remained inferior to that of the ASOMA-ACDF. When the SNR reached −11 dB, the root-mean-square error of the ASOMA-ACDF decreased to 0.22 ms, while that of the ASOMA decreased to 0.32 ms, and that of the A-PSR was approximately 0.42 ms.

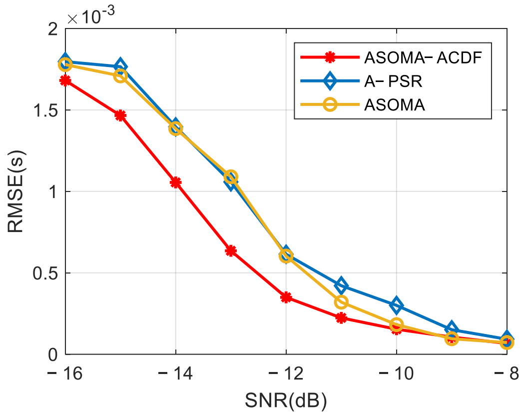


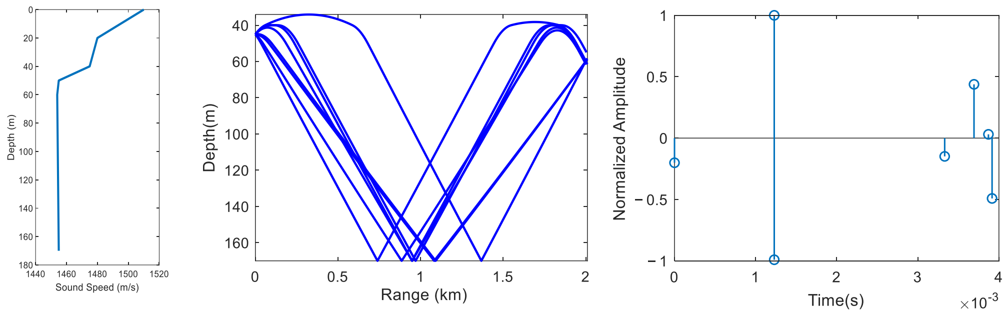
Figure 20. Variation in root-mean-square errors with SNR when frequency was imperfectly estimated.

4.4. Multipath Effect on Estimation Performance

In this experiment, the source information codes were randomly generated, and the PN sequence was a seventh-order m-sequence. The averaging method, with five accumulations, was applied to the ASOMA, A-PSR, and ASOMA-ACDF.

(1) Shallow-sea channels

Assuming a sound source depth of 45 m, a receiver depth of 60 m, and a distance of 2 km, the sound speed profile is shown in Figure 21a, the eigenray is shown in Figure 21b (calculated via "Bellhop"), and the impulse response is shown in Figure 21c (calculated via "Bellhop").



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

Figure 21. Characteristics of shallow-sea channel: (**a**) sound speed profile; (**b**) eigenray; (**c**) impulse response.

The performances of the three methods under the shallow-sea channel are compared in Figures 22 and 23.

Figures 22 and 23 shows that when a seventh-order m-sequence served as the spreading sequence, the estimation performances of all three methods improved with the increase in the SNR. Among them, the ASOMA-ACDF achieved the best performance, followed by the ASOMA, while the A-PSR was inferior to the ASOMA. Specifically, when the SNR was −13 dB (Figure 22), the correct estimation probability of the ASOMA-ACDF was 0.934, compared with 0.742 for the ASOMA and 0.594 for the A-PSR. When the SNR was

−12 dB, the root-mean-square error of the ASOMA-ACDF decreased to 0.05 ms, while that of the ASOMA decreased to 0.1495 ms, and that of the A-PSR was approximately 0.2284 ms (Figure 23).

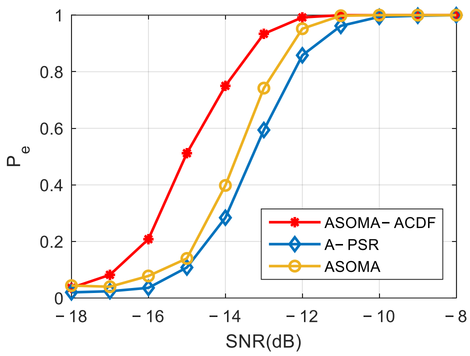


Figure 22. Variation in correct estimation probabilities with SNRs under shallow-sea channel.

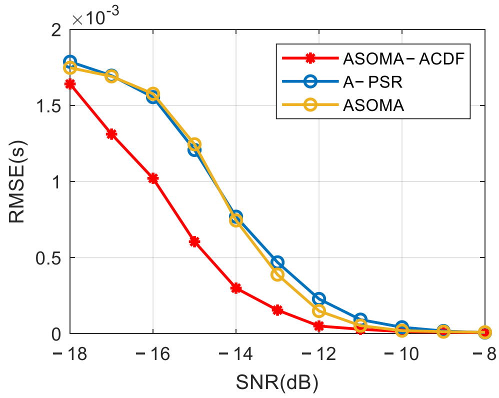


Figure 23. Variation in root-mean-square errors with SNRs under shallow-sea channel.

(2) Deep-sea channels

Assuming a sound source depth of 500 m, a receiver depth of 200 m, and a distance of 15.5 km, the sound speed profile is shown in Figure 24a, the eigenray is shown in Figure 24b (calculated via "Bellhop"), and the impulse response is shown in Figure 24c (calculated via "Bellhop").

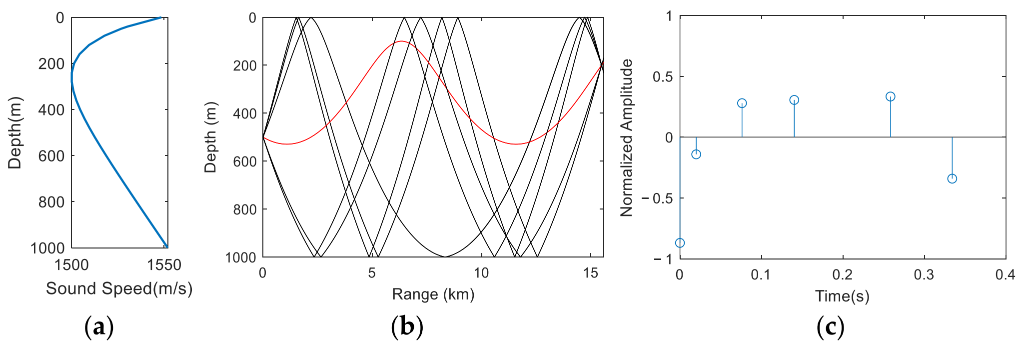


Figure 24. Characteristics of deep-sea channel: (**a**) sound speed profile; (**b**) eigenray; (**c**) impulse response.

The performances of the three methods under the deep-sea channel are compared in Figures 25 and 26.

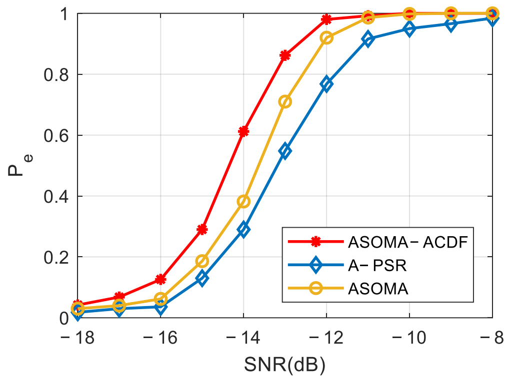


Figure 25. Variation in correct estimation probabilities with SNR under deep-sea channel.

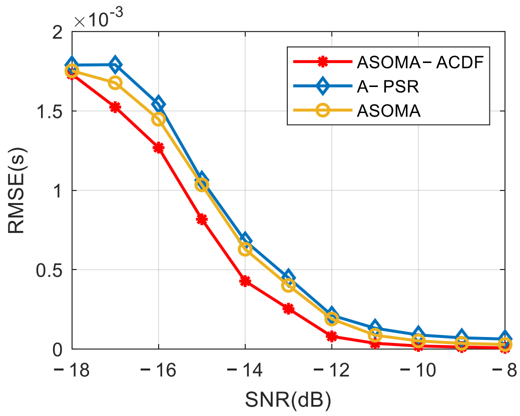


Figure 26. Variation in root-mean-square errors with SNR under deep-sea channel.

Figures 25 and 26 show that when a seventh-order m-sequence served as the spreading sequence, the estimation performances of all three methods improved with the increase in the SNR. Among them, the ASOMA-ACDF achieved the best performance, followed by the ASOMA, while the A-PSR was inferior to the ASOMA. Specifically, when the SNR was −13 dB (Figure 25), the correct estimation probability of the ASOMA-ACDF was 0.862, compared with 0.71 for the ASOMA and 0.548 for the A-PSR. When the SNR was −11 dB, the root-mean-square error of the ASOMA-ACDF decreased to 0.036 ms, while that of the ASOMA decreased to 0.087 ms, and that of the A-PSR was approximately 0.132 ms (Figure 26).

4.5. Complexity Comparison

The complexities of the three methods were compared assuming that the signal was divided into M groups, the number of PN sequence periods contained in each signal group was N each PN code period contained L chips, and each chip contained N_c samples, and the results are shown in Table 1.

Table 1. Comparison of A-PSR, ASOMA, and ASOMA-ACDF complexities.

Estimation Method	Complexity
A-PSR	$O(MLNN_c \log LNN_c)$
ASOMA	$O\left(ML^2N^2N_c^2\right)$
ASOMA-ACDF	$O\left(MLNN_c^2 + ML^2N^2\right)$

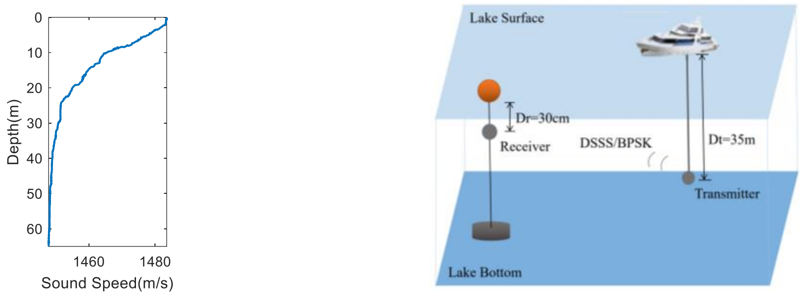
Table 1 shows that the ASOMA-ACDF complexity is lower than that of the ASOMA. Although it is slightly larger than that of the A-PSR, the complexity of the ASOMA-ACDF is completely acceptable in practical applications.

5. Measured Data

5.1. Performance Comparison

This experiment was conducted at Qiandao Lake in Hangzhou, Zhejiang Province. The depths of the transmitter and receiver were 35 m and 30 cm, respectively. The transmitter emitted a short-code direct-sequence spread spectrum signal using BPSK subcarrier modulation. The carrier frequency was set at 10 kHz, with a chip width of 0.3379 ms. The PN sequence utilized was a fifth-order m-sequence, and the PN sequence period of the signal was 10.475 ms. The receiver had a bandwidth ranging from 4 to 16 kHz and a sampling rate of 110 kHz.

The sound speed profile and experimental setup are shown in Figure 27.

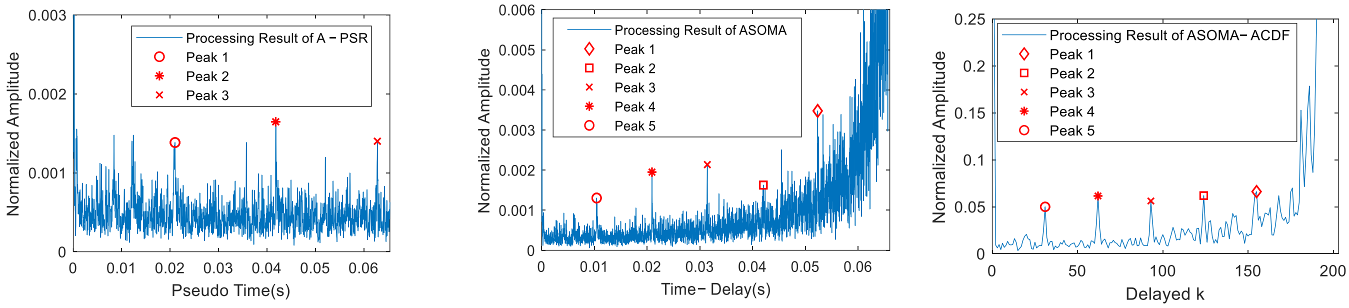


(**a**) (**b**)

Figure 27. Experimental environment: (**a**) sound speed profile; (**b**) experimental setup.

In this experiment, each frame contained 90 information codes, and 1000 frames of signals were observed in total.

The results of the A-PSR, ASOMA, and ASOMA-ACDF processing of one frame when the ensemble averaging count was 15 are shown in Figure 28.



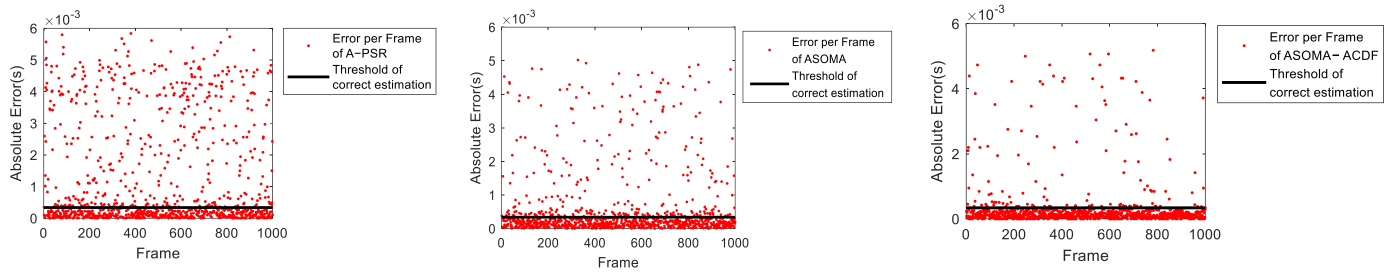
(**a**) (**b**) (**c**)

Figure 28. Peak performances of three methods processing one frame: (**a**) A-PSR; (**b**) ASOMA; (**c**) ASOMA-ACDF.

Figure 28 shows the results of the performances of the three methods in the processing of one frame. All three methods estimated the PN sequence period by extracting the peak intervals; thus, the peak performance is critical for the estimation (the correct peaks to be extracted are marked).

Figure 28a shows the result after A-PSR processing, wherein there are fake peaks with similar amplitudes around the peaks to be extracted, which makes it difficult to correctly extract peak intervals. Figure 28b shows the result after ASOMA processing, where the peaks to be extracted are clearer compared with those of the A-PSR. However, the height of Peak 2 is relatively low, and there is a higher fake peak beside it, which makes it difficult to correctly extract the Peak 2 position and is detrimental to peak interval estimation. Figure 28c shows the ASOMA-ACDF processing result, where the heights of the peaks to be extracted are similar and clearer with no peak missing, which is beneficial for peak interval extraction.

Using the three methods to process all 1000 frames of data, the estimation errors for each frame were obtained and are illustrated in Figure 29.



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

Figure 29. Absolute estimation errors of three methods for each frame: (**a**) A-PSR; (**b**) ASOMA; (**c**) ASOMA-ACDF.

Figure 29 shows the absolute estimation errors of the three methods for each frame (a total of 1000 frames) and a benchmark representing one chip width. According to the figure, the A-PSR has the most outliers, with the least number of estimation errors below the benchmark; the ASOMA has fewer outliers than the A-PSR and more errors below the benchmark; the ASOMA-ACDF has the fewest outliers and the largest number of estimation errors below the benchmark.

Using the 1000 estimation errors shown in Figure 29 as samples, the correct estimation (i.e., absolute error below the benchmark) probability and RMSE (as defined in Section 4) were calculated to compare the performances of the three methods, as shown in Table 2.

Table 2. Comparison of performances of three methods for fifth-order m-sequence.

Estimation Method	Correct Estimation Probability (%)	Root-Mean-Square Error (ms)
A-PSR	53.2	0.23
ASOMA	67.8	0.21
ASOMA-ACDF	84.8	0.17

As shown in Table 2, the ASOMA-ACDF achieved the highest accuracy, the ASOMA exhibits lower correct estimation probabilities than the ASOMA-ACDF, while the A-PSR has the lowest probability. Regarding the RMSE, the ASOMA-ACDF outperformed the others with the smallest RMSE, and the ASOMA method showed a better RMSE performance than that of the A-PSR.

Among the three methods, the A-PSR has the lowest complexity but performs worse than the other two methods. Although the ASOMA performs better than the A-PSR, it is inferior to the ASOMA-ACDF and has the highest complexity. Although the ASOMA-ACDF has a higher complexity than that of the A-PSR, it is completely acceptable in practical applications and has the best estimation performance. Therefore, the ASOMA-ACDF allows for the best performance–complexity tradeoff.

5.2. Multipath Effect Analysis

In this experiment, the experimental environment was relatively stable, the lake surface was calm, and the multipath effect was not severe. All three methods worked properly in these multipath channels. Therefore, this method has a certain robustness to multipath effects. However, the performances of the three methods will all sharply decline in severe multipath and unstable environments. In conclusion, the proposed method has advantages in common scenarios with low SNRs and mild multipath effects.

6. Conclusions

To achieve the high-accuracy estimation of the PN sequence period for DSSS signals under low-SNR conditions, a novel estimation method based on the autocorrelation characteristics of the accumulated code difference function is proposed in this study. The simulation analysis indicates that (1) the proposed method maintains an effective estimation capability under low-SNR conditions, exhibiting a higher correct estimation probability compared with ASOMA and A-PSR; (2) a superior estimation accuracy is achieved with a smaller root-mean-square error than those of the ASOMA and A-PSR; (3) the proposed method has a certain robustness to multipath effects. Finally, lake experiments demonstrated that the proposed method successfully accomplished PN sequence period estimation and exhibited significant advantages compared with the A-PSR and ASOMA under low-SNR conditions.

Author Contributions: Conceptualization, Y.T. and Y.W.; data curation, Y.W.; formal analysis, Y.T. and J.F.; funding acquisition, Y.H.; investigation, Y.X.; methodology, Y.T.; project administration, G.L.; resources, G.L.; software, Y.T.; supervision, G.L.; validation, J.F.; visualization, J.W.; writing—original draft, Y.T.; writing—review and editing, Y.T. and J.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Young Elite Scientists Sponsorship Program by CAST under Grant number 2023QNRC001.

Data Availability Statement: The data presented in this study are not publicly available due to privacy and confidentiality restrictions.

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

References

1. Marques da Silva, M.; Dinis, R.; Aleixo, J.; Oliveira, L.M.L. On the Performance of LDPC-Coded MIMO Schemes for Underwater Communications Using 5G-like Processing. *Appl. Sci.* **2022**, *12*, 5549. [CrossRef]
2. Le Nir, V.; Scheers, B. Robust blind carrier frequency synchronisation for direct sequence spread spectrum systems. *Electron. Lett.* **2015**, *51*, 425–427. [CrossRef]
3. Burel, G.; Bouder, C. Blind estimation of the pseudo-random sequence of a direct sequence spread spectrum signal. In Proceedings of the 21st Century Military Communications Conference 2000, Los Angeles, CA, USA, 22–25 October 2000.
4. Wang, B.; Shen, L.; Wang, H. Direct sequence spread spectrum (DSSS) signal detection based on eigenvalues local binary pattern residual network (EL-ResNet). *Signal Image Video Process.* **2024**, *18*, 4741–4751. [CrossRef]
5. Sun, D.; Wu, J.; Zheng, H.C. The line-of-sight peak detection and tracking of underwater acoustic DSSS communications in the doubly spread channel. *Appl. Acoust.* **2024**, *216*, 109761.1–109761.10. [CrossRef]
6. Zhou, F.; Yin, Y.L.; Qiao, G. Burst mode spread spectrum technology for covert underwater acoustic communication. *Acta Acust.* **2017**, *42*, 37–47.
7. Lu, M.; Cheng, F.; Wei, S. Comparison of detection methods for non-cooperative underwater acoustic DSSS signals. In Proceedings of the IEEE International Conference on Signal Processing, Communications and Computing, Xiamen, China, 22–25 October 2017; pp. 1–5.
8. He, J.A.; Chen, X. Parameter Estimation Technology and Its Development of Communication. *Signals Meas Control.* **2017**, *36*, 6–10.
9. Mehboodi, S.; Jamshidi, A.; Farhang, M. Two algorithms for spread-spectrum sequence estimation for DSSS signals in non-cooperative communication systems. In Proceedings of the 2016 24th Iranian Conference on Electrical Engineering (ICEE), Shiraz, Iran, 10–12 May 2016.

10. Zhang, Z.; Lei, J. A Detecting Algorithm of DSSS Signal Based on Auto-correlation Estimation. In Proceedings of the 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference, Chongqing, China, 25–26 March 2017; pp. 137–141.
11. Sheng, S.; Yang, W.; Hou, Y. An Improved Power Spectrum Reprocessing Method for DS-SS Signal on Spreading Code Period Estimation. In Proceedings of the 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference, Xi'an, China, 25–27 May 2018; pp. 774–779.
12. Feng, C.; Tao, S.; Fan, Z. Analysis on an Improved Pseudo-Code Periodic Estimation Method for DSSS Signals. In Proceedings of the 2nd International Conference on Intelligent Information Processing, Bangkok, Thailand, 17–18 July 2017.
13. Miao, P.; Zhang, T.; Zhang, W.; Ma, G.; Liu, Y. Period Estimation of the PN Sequence for Direct Sequence Spread Spectrum in Multipath Environment. In Proceedings of the 5th International Conference on Wireless Communications, Networking and Mobile Computing, Beijing, China, 24–26 September 2009; pp. 1–4.
14. Gu, H.Q.; Liu, X.X.; Xu, L.; Zhang, Y.J.; Lu, Z.M. Period Estimation of Spread Spectrum Codes Based on ResNet. *Sensors* **2023**, *23*, 7002. [CrossRef]
15. Bai, J.; Zhang, T.; Yu, X.; Wang, Y.E. The estimation of the PN sequence's period of the DSSS signals in narrowband interference environment. In Proceedings of the 2011 IEEE International Conference on Signal Processing, Communications and Computing, Xi'an, China, 14–16 September 2011; pp. 1–4.
16. Burel, G.; Bouder, C.; Berder, O. Detection of direct sequence spread spectrum transmissions without prior knowledge. In Proceedings of the 2001 IEEE Global Telecommunications Conference, San Antonio, TX, USA, 25–29 November 2001; pp. 236–239.
17. Zhang, T.Q.; Yang, L.F.; Dai, S.S.; Li, X.S. Estimation of period and chip rate of pseudo-noise sequence for direct sequence spread spectrum signals using correlation techniques. *Aeospace Electron. Warf.* **2007**, *23*, 53–57. (In Chinese)
18. Zhang, T.Q.; Zhou, Z.Z. A new spectral method of periodic detection of PN sequence in lower SNR DS/SS signals. *Chin. J. Radio Sci.* **2001**, *16*, 518–521. (In Chinese)
19. Yu, Z.; Hao, W.; Yong, C. Period Estimation of PN Sequence for Weak DSSS Signals Based on Improved Power Spectrum Reprocessing in Non-cooperative Communication Systems. In Proceedings of the 2012 International Conference on Control Engineering and Communication Technology, Shenyang, China, 7–9 December 2012; pp. 924–927.
20. Shen, Z.H.; Tang, B.; Lv, Y. Multiple parameters estimation simultaneously for DS-SS/BPSK signal based on fourth-order cumulant 2D slice. In Proceedings of the 2004 International Conference on Communications, Circuits and Systems, Chengdu, China, 27–29 June 2004; pp. 867–870.
21. Kuehls, J.F.; Geraniotis, E. Presence detection of binary-phase-shift-keyed and direct-sequence spread-spectrum signals using a prefilter-delay-and-multiply device. *IEEE J. Sel. Areas Commun.* **1990**, *8*, 915–933. [CrossRef]
22. Dong, Z.Q.; Hu, H.Y.; Yu, H.Y. The Detection, Symbol Period and Chip Width Estimation of DSSS Signals Based on Delay-Multiply, Correlation and Spectrum Analysis. *J. Electron. Inf. Technol.* **2008**, *30*, 840–842. [CrossRef]
23. Zhang, X.L.; Li, S.J.; Chen, Z.M. A DSSS Signal Detection Method Based on Wavelet Decomposition and Delay Multiplication. In Proceedings of the 2018 IEEE Asia-Pacific Conference on Antennas and Propagation, Auckland, New Zealand, 5–8 August 2018; pp. 341–343.
24. Zhong, S.H.; Yu, X.P.; Hou, Y.; Jia, L.; Liu, J.; Yang, J.F. Research on the method of pseudo code period estimation for DSSS signals based on improved cepstrum. *Inf. Technol.* **2021**, *7*, 98–102+108. (In Chinese) [CrossRef]
25. Shockley, W. The theory of p-n junctions in semiconductors and p-n junction transistors. *Bell Labs Tech. J.* **2013**, *28*, 435–489. [CrossRef]
26. Shen, C.; Gao, W.; Song, Z.X.; Yao, T.; Li, B. Parameter estimation of digital communication signal. In Proceedings of the 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications, Ottawa, ON, Canada, 29–30 September 2014; pp. 1080–1083. [CrossRef]