

Research Article

A Sparse Temporal Synchronization Algorithm of Laser Communications for Feeder Links in 5G Nonterrestrial Networks

Lichen Zhu , Hangcheng Han , Xiangyuan Bu , and Jichao Wang 

School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China

Correspondence should be addressed to Hangcheng Han; hanhangcheng@bit.edu.cn

Received 26 January 2018; Accepted 1 May 2018; Published 20 June 2018

Academic Editor: Nan Yang

Copyright © 2018 Lichen Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To foster the rollout of 5G in unserved areas, 3GPP has kicked off a study item on new radio to support nonterrestrial networks (NTNs). Due to ultra-wideband of laser, laser communication is very promising for the feeder links of NTNs; however, imprecise temporal synchronization hinders its deployment, which results from a combination of propagation delay, velocity, acceleration, and jerk of NTN platform. The prior synchronization algorithms are inapplicable to the temporal synchronization in laser communications due to the extremely high data rate and Doppler shift. This paper is devoted to addressing the temporal synchronization problem in laser communications. In particular, we first observe the sparsity of laser signal in time-frequency domain. On top of this observation, we propose a new sparsity-aware algorithm for temporal synchronization without carrier aid through sparse discrete polynomial-phase transform and sparse discrete fractional Fourier transform. Subsequently, we implement the proposed algorithm via designing a hardware prototype. To further evaluate its performance, we conduct extensive simulations, and the results demonstrate the effectiveness of the proposed algorithm in terms of good accuracy, low power consumption, and low computational complexity, suggesting its attractiveness for the feeder links of 5G NTNs.

1. Introduction

With the exponential growth of wireless traffic volume, new radio (NR) becomes the foundation of 5G to provide universal coverage [1, 2]. To foster the rollout of 5G in unserved areas, nonterrestrial networks (NTNs) have been kicked off by 3GPP [3–5]. In this context, the major challenge is that the feeder links between the gateway and the NTN platforms must be broadband; i.e., the data rate should be on the order of gigabits per second (Gbps) [6–8]. Fortunately, laser is innately of the ultra-wideband, and the laser communication is thus very promising for the feeder links in satisfying the data rate requirement, while it is difficult for laser communications to achieve precise temporal synchronization because of the high data rate, e.g., the order of Gbps, and the high Doppler resulting from the movement of NTN platforms [9, 10].

To address the temporal synchronization problem, a large body of works have been proposed, which can

be classified into three categories: tracking loop-based methods, transformation-domain methods, and sparse transformation-domain methods.

Specifically, tracking loop is a type of conventional temporal synchronization algorithms [11–14]. In [11], the authors developed a novel architecture for tracking loop to accurately measure the aircraft's speed, which is helpful for temporal synchronization. In [12, 13], the performance of tracking loop on ultra-wideband impulse radio was analyzed. A common tracking loop called digital delay locked loop (DDLL) was discussed in [14] for tracking direct sequence spread spectrum signal with high Doppler. As analyzed in these works, tracking loop algorithms are suitable for systems with low Doppler and low data rate. While the convergence rate degrades rapidly with the increase of Doppler and Doppler changing rate. Furthermore, high data rate system requires parallel architecture of tracking loop, which needs more resources than the serial one because multiple samples have to be handled during one clock cycle [15, 16] and

cannot meet the low power consumption requirement on NTN platforms [17].

The transformation-domain algorithms [18–21] follow open loop principle and are able to maintain a constant processing delay in high Doppler environments. In [18], the discrete fractional Fourier transform (DFrFT) was adopted for acceleration and velocity estimation. In [19], the discrete polynomial-phase transform (DPT) was used to estimate the aircraft dynamic parameters. In [20], the authors studied the time-frequency characteristics from the perspective of signal time-frequency distribution, i.e., Wigner-Ville distribution, to facilitate the temporal synchronization. In [21], the keystone transform was employed to solve the target range migration problem in temporal synchronization. Although transformation-domain algorithms are suitable for the scenario with high Doppler, their computational complexity is usually very high, which violates the low power consumption requirement as well.

To reduce the complexity, the sparse transformation-domain algorithm is introduced into high Doppler and high data rate systems [22–25]. The authors in [22] proposed a sparse algorithm based on the fast Fourier transform (FFT), namely, sparse FFT (SFFT). The applications of SFFT were introduced in [23], and the authors in [24] further evaluated the implementation performance of SFFT. The paper [25] designed a sparse using the discrete fractional Fourier transform (DFrFT) for the temporal synchronization in the scenario with high Doppler changing rate. With low complexity, the sparse algorithm is recommended to low power consumption platforms. However, these sparse transformation-domain algorithms require carrier recovery. It is pointed out that carrier recovery is difficult to implement on laser communications for NTN platforms because of the complex structure [26] and then restricts the application of the sparse transformation-domain algorithm on the feeder links of 5G NTNs.

Motivated from the observations above, we employ an incoherent laser transmission system called intensity modulation direct detection (IMDD) for its simplicity and low cost [27, 28]. The IMDD system can be modeled as a Gaussian channel with the positive real input signal and input-dependent noise [26, 29]. This paper designs a sparse transformation-domain algorithm in IMDD-based laser communications for 5G NTNs, which is of good accuracy, low power consumption, and low computational complexity. The primary contributions of the paper are summarized as follows.

- (1) We develop a temporal synchronization algorithm based on the sparse pilot. Armed with the sparse DPT (SDPT) and the sparse DFrFT (SDFrFT), the proposed algorithm is able to estimate the propagation delay, velocity, acceleration, and jerk without carrier aid in high dynamic environment.
- (2) We analyze the accuracy and complexity of the proposed algorithm and compare its performance with the non-sparse algorithms and the DLL algorithms. The analytical and experimental results show that

the proposed algorithm performs similar to the non-sparse algorithm and superior to the DLL algorithm in terms of accuracy. While the proposed algorithm can achieve lower complexity, making it the most suitable for scenarios with high Doppler and low power consumption among these three algorithms.

- (3) We discuss the implementation issues including module reuse analysis and clock rate analysis. According to the analysis, the proposed algorithm can be implemented with less clock rate than the conventional DLL algorithm.
- (4) We implement the proposed algorithm via designing a hardware prototype, and the results agree well with the theoretical analysis, demonstrating that the algorithm can be implemented on resource constrained platforms, e.g., 5G NTN platforms.

The rest of the paper is organized as follows. In Section 2, the system model is described. The proposed temporal synchronization algorithm is then presented in Section 3 and analyzed in Section 4. The related implementation issues are shown in Section 5. Simulation and experiment results are given in Sections 6 and 7 to demonstrate the effectiveness of the proposed algorithm, respectively, which are followed by the conclusions drawn in Section 8.

2. System Model

The system model of NTNs is presented in Figure 1, where the IMDD-based laser communications serve for the feeder link between spaceborne platform and gateway.

The laser signal is received by photodetection and is transformed to an electrical signal with positive and negative level, which is expressed as

$$\begin{aligned} s(t) &= A_0 g(t - \tau(t)) + w(t) \\ &= A_0 g\left(t - \left(a_0 + a_1 t + a_2 t^2 + a_3 t^3\right)\right) + w(t), \end{aligned} \quad (1)$$

where $g(t)$ is the waveform of signal with amplitude A_0 , $\tau(t)$ denotes the propagation delay, which contains four components related to distance (a_0), velocity (a_1), acceleration (a_2), and jerk (a_3), respectively, and $w(t)$ is the zero-mean additive white Gaussian noise with variance σ^2 . The experimental results show that $a_1 \gg a_2 \gg a_3$ [11]. The nature of temporal synchronization is to estimate $\tau(t)$, namely, a_0 , a_1 , a_2 , and a_3 , which can be obtained with the aid of pilot.

Figure 2 shows the pilot position in the transmitted frame, where L_{frm} , L_h , and L_p denote the lengths of the entire frame, the frame header, and the pilot, respectively. The pilot waveform, denoted by $g_p(t)$, is a sequence of periodic square waves and is expressed as

$$g_p(t) = \begin{cases} \frac{1}{2}, & kT_{syb} < t \leq \left(k + \frac{1}{2}\right)T_{syb}, \\ -\frac{1}{2}, & \left(k + \frac{1}{2}\right)T_{syb} < t \leq (k + 1)T_{syb}, \end{cases} \quad (2)$$

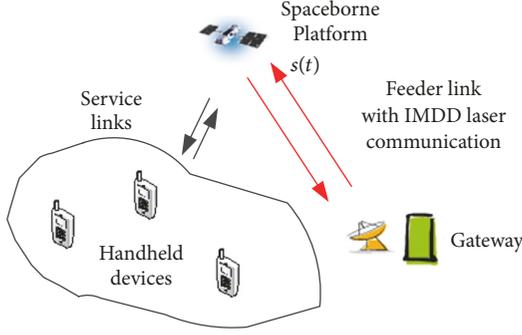


FIGURE 1: The system model of NTN.

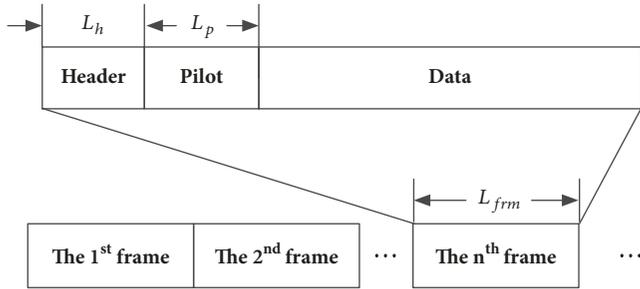


FIGURE 2: The pilot in the transmitted frame.

where T_{syb} denotes the symbol duration and $k = 0, 1, 2, \dots, L_p - 1$. According to the Fourier series, we rewrite (2) as

$$g_p(t) = \sum_{k=1}^{\infty} \frac{1}{2k-1} \sin(2(2k-1)\pi f_c t), \quad (3)$$

where $f_c = 1/(2T_{syb})$. Through low-pass filtering, (3) is converted to

$$g_p(t) = \sum_{k=1}^{\infty} \frac{A_k}{2k-1} \sin(2(2k-1)\pi f_c t), \quad A_k \in (0, 1), \quad (4)$$

where A_k denotes the amplitude of the k th order harmonic, which decreases rapidly with the increase of k .

After analog-digital conversion, from (1) and (4), we have

$$\begin{aligned} s(n) &= r(n) + w(n) \\ &= A_0 g(nT_s - \tau(nT_s)) + w(n), \end{aligned} \quad (5)$$

$$g_p(nT_s) = \sum_{k=1}^{\infty} \frac{A_k}{2k-1} \sin(2(2k-1)\pi f_c nT_s),$$

where T_s denotes the sample interval.

To simplify the computation, we assume $A_k = 0$ when $k > 3$; then $g_p(nT_s)$ in (5) is rewritten as

$$g_p(nT_s) = \sum_{k=1}^3 \frac{A_k}{2k-1} \sin(2(2k-1)\pi f_c nT_s). \quad (6)$$

3. The Proposed Temporal Synchronization Algorithm

As $g_p(nT_s)$ is the sum of sinusoidal functions, the parameters estimation can be expressed as follows according to the maximum likelihood principle [30, 31],

$$\begin{aligned} & \{\hat{a}_1, \hat{a}_2, \hat{a}_3\} \\ &= \operatorname{argmax}_{a_1, a_2, a_3} \left| \sum_n s_c(n) e^{-j2\pi((f_c - a_1)nT_s - a_2(nT_s)^2 - a_3(nT_s)^3)} \right|, \quad (7) \\ & \hat{a}_0 = \operatorname{arg} \left(\sum_n s_c(n) e^{-j2\pi((f_c - \hat{a}_1)nT_s - \hat{a}_2(nT_s)^2 - \hat{a}_3(nT_s)^3)} \right), \end{aligned}$$

where $s_c(n) = s(n) + j \cdot s_{\pi/4}(n)$, and $s_{\pi/4}(n)$ represents $s(n)$ with $\pi/4$ phase shift. Based on (7), we will elaborate how to perform the temporal synchronization in four steps, as shown in Figure 3. We conduct the preprocess in the first step to facilitate the following three steps. In the second step, as estimating \hat{a}_3 is difficult within limited process duration in high Doppler environment, the parameter \hat{a}_2 will be preferentially derived based on a two-order SDPT (SDPT₂). In the third step, the parameters \hat{a}_0 and \hat{a}_1 will be obtained based on SDFrFT, and \hat{a}_2 will also be improved accordingly. The parameter \hat{a}_3 will be presented based on a three-order SDPT (SDPT₃) after resampling in the fourth step. Finally, we output all the estimated results, namely, $\hat{\tau}(nT_s)$.

3.1. The First Step: Preprocess. To prepare data for estimation, preprocess contains three stages.

First, the pilot is located with the aid of frame header. Based on the cross-correlation between the received signal and the template in header, the pilot is located by

$$\begin{aligned} \hat{n}_p &= \operatorname{argmax}_n \left(\sum_{b=0}^{\lfloor f_s L_h T_{syb} \rfloor - 1} s(n) \cdot g_h((n-b)T_s) \right) \\ &+ \lfloor f_s L_h T_{syb} \rfloor, \end{aligned} \quad (8)$$

where $\lfloor \cdot \rfloor$ denotes the floor function, $f_s = 1/T_s$ denotes the sample rate, and $g_h(nT_s)$ is the header waveform.

Second, we derive the complex-form of $s(n)$ to meet the requirement of (7). Considering the $\pi/4$ phase shift of $g_p(nT_s)$, namely,

$$\begin{aligned} g_{p-\pi/4}(nT_s) &= g_p\left(nT_s - \frac{T_{syb}}{2}\right) \\ &= \sum_{k=1}^3 \frac{A_k}{2k-1} \sin\left(2(2k-1)\pi f_c \left(nT_s - \frac{T_{syb}}{2}\right)\right) \\ &= \sum_{k=1}^3 \frac{(-1)^{k-1} A_k}{2k-1} \cos(2(2k-1)\pi f_c nT_s), \end{aligned} \quad (9)$$

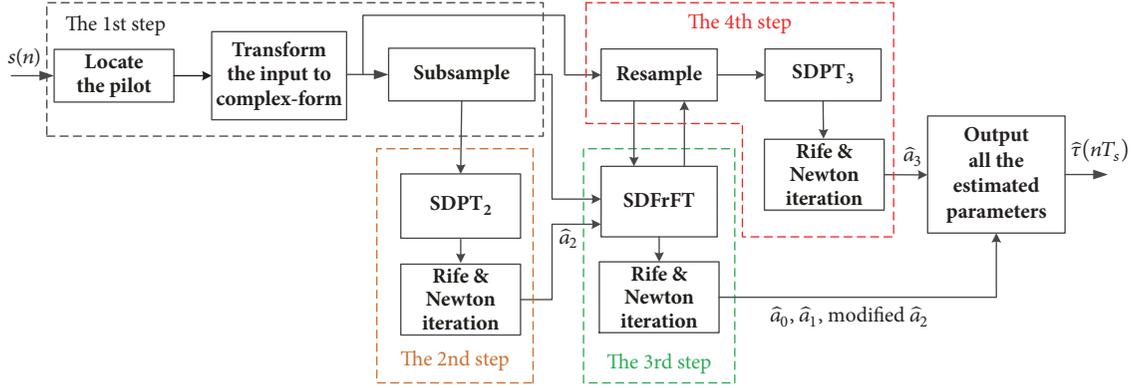


FIGURE 3: The diagram of the sparsity-based scheme.

and the complex-form of $g_p(nT_s)$ is expressed as

$$\begin{aligned} g_c(nT_s) &= g_{p-\pi/4}(nT_s) + j \cdot g_p(nT_s) \\ &= \sum_{k=1}^3 \frac{(-1)^{k-1} A_k}{2k-1} \exp((-1)^{k-1} j 2(2k-1)\pi f_c nT_s) \quad (10) \\ &= \mathbf{A}^T \mathbf{P}, \end{aligned}$$

where $\mathbf{A} = [A_1, -A_2/3, A_3/5]^T \in \mathbb{R}^3$, and $\mathbf{P} = [P_1, P_2, P_3]^T \in \mathbb{C}^3$ with $P_k = \exp(j(-1)^{k-1} 2(2k-1)\pi(f_c nT_s - \tau(nT_s)))$ and $k = 1, 2, 3$. From (10), the complex-form of $s(n)$ is

$$s_c(n) = A_0 g_c(nT_s) + w(n) = \mathbf{A}_0^T \mathbf{P} + w(n), \quad (11)$$

where $\mathbf{A}_0 = [A_0 A_1, -A_0 A_2/3, A_0 A_3/5]^T \in \mathbb{R}^3$.

Third, since the pilot has been located, we obtain N_{seq} entries by subsampling the pilot data start from \hat{n}_p with subsample rate $r_{smp} = \beta L_{frm} T_{syb} / T_s$, where β is a positive integer. The challenge here lies in subsampling. The subsequent samples will drift away from the original position caused by Doppler, and the sample position may be out of the pilot range. Hence, a sufficient length of pilot, i.e., L_p , is required to prevent the sampling position out of range. In the following, we discuss the minimum value of L_p .

Letting L_m denote the maximum range of drift, we have

$$L_m = N_{seq} \frac{|a_{1m}| \beta L_{frm}}{c}, \quad (12)$$

where a_{1m} denotes the maximum velocity. The reason for ignoring a_2 and a_3 is that the platform cannot accelerate any more after it reaches the maximum velocity. Generally, as the drift direction is unknown, the lower limit of L_p is $\lceil 2L_m \rceil$, where $\lceil \cdot \rceil$ denotes the ceiling function. An improved method can reduce the lower limit of L_p to $\lceil L_m \rceil$ by

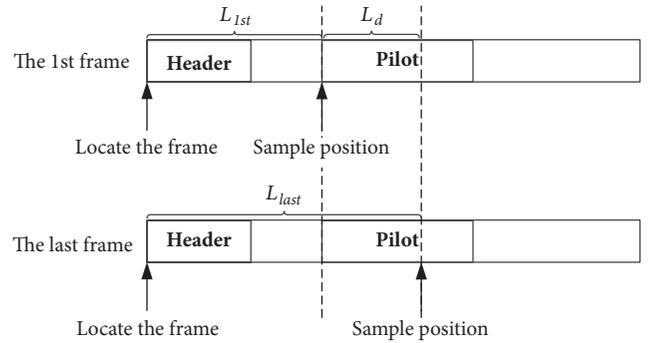


FIGURE 4: The method of determining drift direction and magnitude.

simultaneously obtaining 3 sample sets that start from \hat{n}_p , $\hat{n}_p + (\lceil L_m \rceil / 2) f_s T_{syb}$, and $\hat{n}_p + \lceil L_m \rceil f_s T_{syb}$, respectively, among which at least a set will not be out of range. Then we select the best one among these sets with the following method. Since the frame header is exploited to locate the frame, it assists in determining the drift direction and magnitude, as shown in Figure 4, where L_{1st} and L_{last} denote the intervals from the sample position to the header in the first frame and to the last frame, respectively, and L_d denotes the drift value with $L_d = L_{1st} - L_{last}$. The first, or second, or third sample set is selected when $L_d > L_p/2$, $L_d \in [-L_p/2, L_p/2]$, or $L_d < -L_p/2$, respectively.

After subsampling, $s_c(n)$ is converted to $s_c(m)$ with $m = 0, 1, \dots, N_{seq} - 1$, and the sample rate and sample interval turn into that $f'_s = f_s / r_{smp}$ and that $T'_s = r_{smp} \cdot T_s$, respectively. In the following section, we focus on the signal processing of the proposed temporal synchronization.

3.2. The Second Step: Derive \hat{a}_2 Based on $SDPT_2$. According to (7), it is possible to jointly estimate all the four unknown parameters. Unfortunately, the complexity of joint estimation is unaffordable. To reduce the complexity, we decompose the parameter estimation problem into several subproblems to estimate the parameters one by one, which is essentially the same as the joint estimation because the parameters to be estimated have a weak association with

each other and the separate estimation results in no performance loss. Generally, the highest-order parameter, a_3 , is estimated firstly [32]. However, as mentioned above, high Doppler makes the sample position drift away and leads to a limited process duration. Moreover, as a_3 represents the third derivative of distance, it cannot be accurately estimated in such short duration. In comparison, the acceleration estimation is much easier, and we thus preferentially estimate a_2 .

As a reduced-order method to estimate a_2 , DPT_2 is defined as [32]

$$\begin{aligned} \text{DPT}_2(s_c(m), f, \xi) &\triangleq \text{DFT}(\text{DP}_2(s_c(m), \xi), f) \\ &= \sum_{m=0}^{N_{\text{seq}}-1} \text{DP}_2(s_c(m), \xi) e^{-j2\pi f m T_s'} \end{aligned} \quad (13)$$

where DFT denotes the discrete Fourier transform with rotation factor $e^{-j2\pi f m T_s'}$, ξ is a positive integer, and DP_2 is given by

$$\text{DP}_2(s_c(m), \xi) \triangleq s_c(m) s_c^*(m - \xi), \quad (14)$$

where $(\cdot)^*$ denotes the conjugate operation. For simplicity, let $x(m) = \text{DP}_2(s_c(m), \xi)$, and we have

$$\begin{aligned} x(m) &= s_c(m) s_c^*(m - \xi) \\ &= (r(m) + w(m))(r^*(m - \xi) + w^*(m - \xi)) \\ &= r(m) r^*(m - \xi) + r(m) w^*(m - \xi) \\ &\quad + r^*(m - \xi) w(m) + w(m) w^*(m - \xi), \end{aligned} \quad (15)$$

where

$$\begin{aligned} r(m) r^*(m - \xi) &= (\mathbf{A}_0^T \mathbf{P} \mathbf{P}_\xi^T \mathbf{A}_0) = \mathbf{A}_0^T \mathbf{P} (\mathbf{P}_\xi^*)^T \mathbf{A}_0 \\ &= \mathbf{A}_0^T \begin{bmatrix} P_1 P_1^*(\xi) & P_1 P_2^*(\xi) & P_1 P_3^*(\xi) \\ P_2 P_1^*(\xi) & P_2 P_2^*(\xi) & P_2 P_3^*(\xi) \\ P_3 P_1^*(\xi) & P_3 P_2^*(\xi) & P_3 P_3^*(\xi) \end{bmatrix} \mathbf{A}_0 \end{aligned} \quad (16)$$

$$\begin{aligned} X_\alpha(k) &\triangleq \begin{cases} \sqrt{\frac{\sin \alpha - j \cos \alpha}{M}} e^{j(\cot \alpha/2)(kU_s)^2} \cdot \sum_{m=0}^{M-1} s_c(m) e^{-j(2\pi k m/M)} e^{j(\cot \alpha/2)(mT_s')^2}, & \alpha \geq 0 \\ \sqrt{\frac{j \cos \alpha - \sin \alpha}{M}} e^{j(\cot \alpha/2)(kU_s)^2} \cdot \sum_{m=0}^{M-1} s_c(m) e^{-j(2\pi k m/M)} e^{j(\cot \alpha/2)(mT_s')^2}, & \alpha < 0 \end{cases} \\ &\approx \sqrt{\frac{1}{M}} e^{j(\cot \alpha/2)(kU_s)^2} \cdot \text{DFT}(s_c(m) e^{j(\cot \alpha/2)(mT_s')^2}). \end{aligned} \quad (19)$$

with $m \in [0, M-1]$ and $k \in [0, M-1]$.

In (19), α and U_s denote the rotation angle and the sample interval of the output, $X_\alpha(k)$, respectively. Besides, $T_s' \times U_s =$

is the principal component of $x(m)$; \mathbf{P}_ξ and $P(\xi)$ denote \mathbf{P} and P with ξ -delay, respectively. The other components of $x(m)$ are related to $w(m)$ and can be regarded as the noise terms. Moreover, $r(m)r^*(m - \xi)$ is dominated by $P_1 P_1^*(\xi)$, $P_2 P_2^*(\xi)$, and $P_3 P_3^*(\xi)$. The amplitude of $P_1 P_1^*(\xi)$ is much larger than that of $P_2 P_2^*(\xi)$ and $P_3 P_3^*(\xi)$. Consequently, the components of $r(m)r^*(m - \xi)$ except for $P_1 P_1^*(\xi)$ are regarded as the disturbance terms, and (15) can be rewritten as

$$\begin{aligned} x(m) &= (A_0 A_1)^2 P_1 P_1^*(\xi) + \ell(m T_s') + \lambda(m T_s') \\ &= (A_0 A_1)^2 e^{j2\pi((-2a_2 \xi T_s'^2 + 3a_3 \xi^2 T_s'^3)m - 3a_3 \xi T_s'^3 m^2 - \phi)} \\ &\quad + \ell(m T_s') + \lambda(m T_s'), \end{aligned} \quad (17)$$

where $\ell(m T_s')$ and $\lambda(m T_s')$ denote the disturbance terms and noise terms, respectively, and ϕ is a constant. Letting $\widehat{X}(k) = \text{DFT}(x(m))$ and $\widehat{k}_0 = \text{argmax}\{|\widehat{X}(k)|\}$, since $3a_3 \xi (T_s')^3 m^2$ and $3a_3 \xi^2 (T_s')^3 m$ are small enough to be ignored, \widehat{a}_2 can be expressed as

$$\widehat{a}_2 = -\frac{1}{2\xi (T_s')^2} \widehat{k}_0. \quad (18)$$

Considering the sparsity of DFT result, SFFT can be employed to reduce the complexity, as shown in Algorithm 1.

As the accuracy of SDPT_2 is limited by the resolution of DFT, which is given by $\Delta f = f_s'/M$, accurately estimating a_2 cannot be achieved with small M . Thus, we adopt Rife and Newton methods to improve the estimation accuracy by updating \widehat{k}_0 and recalculating \widehat{a}_2 due to (18), as shown in Algorithm 2, whose effectiveness is demonstrated in Figure 5.

3.3. The Third Step: Obtain \widehat{a}_0 , \widehat{a}_1 , and the Improved \widehat{a}_2 . In this part, \widehat{a}_0 , \widehat{a}_1 , and the improved \widehat{a}_2 are obtained by Pei sampling-type DFrFT [22], which is given in the following equation:

$(2\pi|\sin \alpha|)/M$ shall be satisfied. Then we derive the optimal values of k and α by

$$\{\widehat{k}_m, \widehat{\alpha}_m\} = \text{arg max}_{\alpha, k} |X_\alpha(k)|. \quad (20)$$

Input: $x(m)$.

Output: $\widehat{X}(k)$.

1: Zero padding and the length of $x(m)$ is changed from N_{seq} to M .

2: Let $L = \log_2 M$.

3: For $i = 0; i < L; i++$ do

4: Tear apart the spectrum randomly by $x_1(m) = x((\sigma_i \cdot m) \bmod M)$ and $X_1(k) = X((\sigma_i^{-1} \cdot m) \bmod M)$, where $X(k)$ and $X_1(k)$ denote the expressions of $x(m)$ and $x_1(m)$ in frequency-domain, respectively, and $m, \sigma_i, k \in [1, M]$.

5: Apply a flat window function to expand the spectrum range.

$$(1) \quad \text{Let } G(k) \in \begin{cases} [1 - \delta, 1 + \delta], & k \in [-\varepsilon' M, \varepsilon' M] \\ [0, \delta], & k \notin [-\varepsilon M, \varepsilon M] \end{cases} \text{ be the window function in frequency-domain,}$$

where $\varepsilon' \in (0, 1)$ and $\varepsilon \in (0, 1)$ denote the cutoff frequencies of the passband and stopband, respectively, and δ denotes the ripple amplitude, whose reference value is $1/M^c$, where c is a positive integer.

$$(2) \quad \text{Compute } x_2(m) = g(m) \cdot x_1(m), \text{ where } g(m) \text{ denotes the expression of } G(k) \text{ in time-domain.}$$

The length of $g(m)$ is $\omega = o(B \log_2(M/\sigma_i))$.

6: Subsample in frequency-domain. Letting B be the data length after subsampling, we have $X_3(k) = \text{FFT}(\sum_{j=0}^{\lfloor M/B-1 \rfloor} x_2(m + j \cdot B))$ with $k \in [1, B]$.

7: Map with a hash function.

$$(1) \quad \text{Define a hash function } h_\sigma(k) = \lfloor \sigma_i \cdot kB/M \rfloor.$$

$$(2) \quad \text{Define an offset function } o_{\sigma_i}(k) = \sigma_i \cdot k - h_{\sigma_i}(k) \cdot B/M.$$

(3) Let Γ_i denote the support set of the largest l coefficients in $X_3(k)$. The preimage set of Γ_i is I_i , whose size is lM/B , where $I_i = \{k \in [1, M] \mid h_{\sigma_i}(k) \in \Gamma_i\}$ and $B = \sqrt{Ml/\log_2(M/\delta)}$.

(4) Obtain the l largest spectrum coefficients as

$$\widehat{X}_{4,i}(k) = \begin{cases} \frac{X_3(h_\sigma(k) e^{-j\pi o_{\sigma_i} k \omega / M})}{G(o_{\sigma_i}(k))}, & k \in I_i, \\ 0, & k \in [1, M] \cap \bar{I}_i. \end{cases}$$

(5) Record the nonzero position of $\widehat{X}_{4,i}(k)$.

8: End for

9: Letting v_k be the occurrence times of coordinate k in the sets and only retaining the coordinate whose occurrence times are larger than $L/2$, we have $I' = \{k \in I_1 \cup \dots \cup I_L \mid v_k > L/2\}$, and the rest terms are zeroes.

10: For each coordinate in I' , we obtain the corresponding spectrum coefficients $\widehat{X}^r(k)$ with $k \in I'$ and $r = 1, 2, \dots, L$.

11: Return $\widehat{X}(k) = \begin{cases} \text{median}\{\widehat{X}^r(k)\}, & k \in I' \\ 0, & k \notin I' \end{cases}$, where $\text{median}(\cdot)$ denotes computing the median of a sequence.

ALGORITHM 1: Estimate \widehat{a}_2 based on the SFFT algorithm.

1: Modify the DFT result by Rife method.

Since $\widehat{X}(k)$ denotes the output of SFFT, the modified result is

$$\widehat{k}_1 = \frac{1}{T_s'} \left[\widehat{k}_0 + r \frac{|\widehat{X}(\widehat{k}_0 + r)|}{|\widehat{X}(\widehat{k}_0)| + |\widehat{X}(\widehat{k}_0 + r)|} \right],$$

where

$$r = \begin{cases} 1, & |\widehat{X}(\widehat{k}_0 + 1)| \geq |\widehat{X}(\widehat{k}_0 - 1)| \\ -1, & |\widehat{X}(\widehat{k}_0 + 1)| < |\widehat{X}(\widehat{k}_0 - 1)|. \end{cases}$$

2: Further modify the result by the iteration of Newton method.

For a positive integer i , the $(i + 1)$ -th iteration result is expressed as $\widehat{k}_{i+1} = \widehat{k}_i - \lambda_i (\widehat{X}'(\widehat{k}_i) / \widehat{X}''(\widehat{k}_i))$, where $\widehat{X}'(\widehat{k})$ and $\widehat{X}''(\widehat{k})$ denote the first-order and the second-order derivatives of $\widehat{X}(\widehat{k})$, respectively, and λ denotes the tuning step.

3: Replace \widehat{k}_0 by the latest iteration result, \widehat{k}_N .

ALGORITHM 2: Modify the DFT result based on Rife and Newton methods.

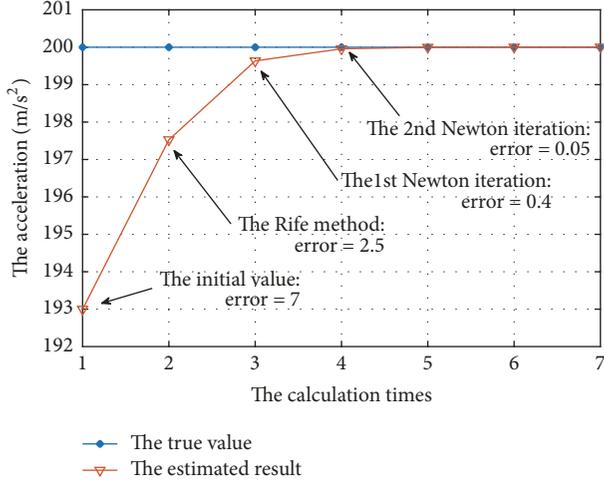


FIGURE 5: The effectiveness of Rife and Newton methods.

From (19) and (20), we obtain the estimated results of \hat{a}_0 , \hat{a}_1 , and \hat{a}_2 in the way

$$\begin{aligned}\hat{a}_2 &= \frac{\cot(\hat{\alpha}_m)}{4\pi}, \\ \hat{a}_1 &= \frac{\hat{k}_m f'_s}{M}, \\ \hat{a}_0 &= \arg \left\{ \frac{X_{\hat{\alpha}_m}(\hat{k}_m)}{-\text{sgn}(\hat{\alpha}_m) e^{j(1/2)[\cot \hat{\alpha}_m (\hat{k}_m U_s)^2 + \hat{\alpha}_m + \pi/2]}} \right\}.\end{aligned}\quad (21)$$

To reduce the computational complexity, we replace DFT in the implementation of Pei sampling-type DFrFT with SFFT due to the sparse spectrum. The implementation of DFrFT based on SFFT is called SDFrFT, which exhibits a similar performance to DFrFT, as shown in Figure 6.

In addition, Rife and Newton methods can be employed again to replace \hat{k}_m with \hat{k}_N , whose procedure is similar to that in Algorithm 2 and is omitted for brevity. Then we substitute \hat{k}_N into (19), (20), and (21) to recalculate \hat{a}_1 and \hat{a}_0 .

3.4. The Fourth Step: Resample and Derive \hat{a}_3 Based on SDPT₃. As mentioned above, high dynamic, especially high velocity brings the drift of sample position and results in an insufficient process duration, which is contrary to accurately estimating a_3 . A method to mitigate the drift by updating the sample rate is introduced as follows.

Note that the velocity, \hat{a}_1 , has been derived from (21); it can be substituted into the following formula to update the sample rate:

$$f_s'' = \frac{[2\hat{a}_1 + R_b] f'_s}{R_b}. \quad (22)$$

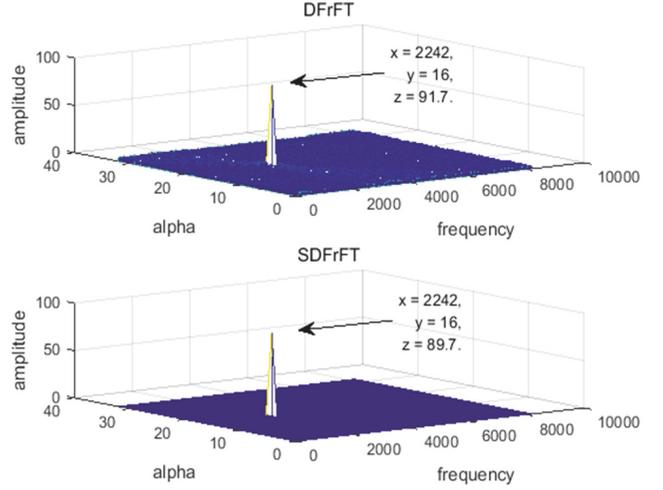


FIGURE 6: The comparison between DFrFT and SDFrFT.

The sample interval turns into that $T_s'' = 1/f_s''$ accordingly. As f_s'' matches with the change of data rate caused by Doppler, the drift of sample position is mitigated, and estimating a_3 becomes easier with the increase of process duration.

In the following, SDPT₃ is introduced to estimate a_3 . Assuming N'_{seq} entries are obtained after resampling, we have

$$\begin{aligned}\text{SDPT}_3(s_c(m), f, \xi) &\triangleq \text{SFFT}(\text{DP}_3(s_c(m), \xi), f) \\ &= \sum_{m=0}^{N'_{seq}-1} \text{DP}_3(s_c(m), \xi) e^{-j2\pi f m T_s''},\end{aligned}\quad (23)$$

where DP_3 is defined as

$$\begin{aligned}\text{DP}_3(s_c(m), \xi) &\triangleq \text{DP}_2(\text{DP}_2(s_c(m), \xi), \xi) \\ &= \text{DP}_2(s_c(m) s_c^*(m - \xi), \xi) \\ &= s_c(m) s_c^*(m - \xi)^2 s_c(m - 2\xi).\end{aligned}\quad (24)$$

Letting $\mathbf{P}_{2\xi}$ and $P(2\xi)$ denote \mathbf{P} and P with 2ξ -delay, respectively, $x_2(m) = \text{DP}_3(s_c(m), \xi)$, and $\lambda'(mT_s'')$ denote the noise terms, (24) can be further expanded as

$$\begin{aligned}x_2(m) &= s_c(m) s_c^*(m - \xi)^2 s_c(m - 2\xi) \\ &= (\mathbf{A}_0^T \mathbf{P}(\mathbf{P}_\xi^*)^T \mathbf{A}_0) \cdot (\mathbf{A}_0^T \mathbf{P}_\xi^* (\mathbf{P}_{2\xi})^T \mathbf{A}_0) \\ &\quad + \lambda'(mT_s'') \\ &= \mathbf{A}_0^T \cdot (\mathbf{P}(\mathbf{P}_\xi^*)^T) \cdot (\mathbf{A}_0 \mathbf{A}_0^T) \cdot (\mathbf{P}_\xi^* (\mathbf{P}_{2\xi})^T) \\ &\quad \cdot \mathbf{A}_0 + \lambda'(mT_s'')\end{aligned}$$

$$\begin{aligned}
&= (\mathbf{A}_0^T \otimes \mathbf{A}_0^T) \cdot (\mathbf{P} (\mathbf{P}_\xi^*)^T \otimes \mathbf{P}_\xi^* (\mathbf{P}_{2\xi})^T) \\
&\quad \cdot (\mathbf{A}_0 \otimes \mathbf{A}_0) + \lambda' (mT_s''),
\end{aligned} \tag{25}$$

where \otimes denotes the Kronecker product. Obviously, $x_2(m)$ is dominated by $P_1 P_1^*(\xi) P_1^*(\xi) P_1(2\xi)$, $P_2 P_2^*(\xi) P_2^*(\xi) P_2(2\xi)$, and $P_3 P_3^*(\xi) P_3^*(\xi) P_3(2\xi)$. Furthermore, as the amplitude of $P_1 P_1^*(\xi) P_1^*(\xi) P_1(2\xi)$ is much larger than that of the other two components, (25) can be rewritten as

$$\begin{aligned}
x_2(m) &= P_1 P_1^*(\xi) P_1^*(\xi) P_1(2\xi) + \ell' (mT_s'') \\
&\quad + \lambda' (mT_s'') \\
&= (A_0 A_1)^4 e^{-j2\pi(6a_3 \xi^2 T_s''^3 m + 2a_2 \xi^2 T_s''^2 - 6a_3 \xi^3 T_s''^3)} \\
&\quad + \ell' (mT_s'') + \lambda' (mT_s''),
\end{aligned} \tag{26}$$

where $\ell' (mT_s'')$ denotes the disturbance terms. From (26), \hat{a}_3 is obtained by

$$\hat{a}_3 = -\frac{1}{6\xi^2 (T_s'')^3} \operatorname{argmax} \{|\operatorname{SFFT}(x_2(m))|\}. \tag{27}$$

The estimation accuracy can be further improved by Rife and Newton methods, whose procedure is similar to that in Algorithm 2 and is omitted for brevity.

In addition, as longer duration is obtained after resampling, we conduct SDFrFT again to further improve the accuracy of \hat{a}_2 , \hat{a}_1 , and \hat{a}_0 , whose procedure is similar to that in the third step and is also omitted for brevity.

4. Performance Analysis

To further present the advantages of the proposed temporal synchronization algorithm, we will theoretically analyze its

performance in terms of estimation accuracy and algorithm complexity in this section.

4.1. Estimation Accuracy Analysis. As \hat{a}_3 , \hat{a}_2 , and \hat{a}_1 are derived from DPT₃, DPT₂, and DFrFT, respectively, we discuss the performance of these three algorithms as follows.

According to (17) and (26), the result of DPT contains the disturbance terms and the noise terms. The ratios of the principal components to the maximum disturbances in DPT₂ and DPT₃ are $9A_1^2/A_2^2$ and $81A_1^4/A_2^4$, respectively. As $A_1 \gg A_2$, the disturbance terms are negligible. The noise terms follow a zero-mean Gaussian distribution with variances $\sigma_{\lambda, \text{DPT}_2}^2$ and $\sigma_{\lambda, \text{DPT}_3}^2$, as shown in (28) and (29), respectively,

$$\begin{aligned}
\sigma_{\lambda, \text{DPT}_2}^2 &= E \left[\left| (\mathbf{A}_0^T \mathbf{P} + w(m)) (\mathbf{A}_0^T \mathbf{P} + w^*(m - \xi)) \right. \right. \\
&\quad \left. \left. - (\mathbf{A}_0^T \mathbf{P})^2 \right|^2 \right] = 2A^2 \sigma^2 + \sigma^4
\end{aligned} \tag{28}$$

$$\begin{aligned}
\sigma_{\lambda, \text{DPT}_3}^2 &= E \left[\left| (\mathbf{A}_0^T \mathbf{P} + w(m)) (\mathbf{A}_0^T \mathbf{P} + w^*(m - \xi))^2 \right. \right. \\
&\quad \left. \left. \cdot (\mathbf{A}_0^T \mathbf{P} + w(m - 2\xi)) - (\mathbf{A}_0^T \mathbf{P})^4 \right|^2 \right] = 6A^6 \sigma^2 \\
&\quad + 10A^4 \sigma^4 + 4A^2 \sigma^6 + \sigma^8
\end{aligned} \tag{29}$$

with $A = A_0(A_1 + A_2/3 + A_3/5)$.

Considering the impact of sample position of pilot signal on the valid signal length, we discuss the SNRs of DPT₃, DPT₂, and DFrFT when the pilot length is insufficient as follows. Ignoring the impact of a_2 and a_3 , the valid signal length is

$$N_{\text{valid}} = \frac{L_p \cdot c}{L_{\text{frm}} \cdot |a_{1m}|}. \tag{30}$$

The rest of signal is approximate random and can be regarded as noise, whose length is $(N_{\text{seq}} - N_{\text{valid}})$. The SNRs of DPT₃, DPT₂, and DFrFT are

$$\begin{aligned}
\text{SNR}_{\text{DPT}_3} &= \frac{N_{\text{valid}}^2 (A_0 A_1)^8}{N_{\text{seq}}' (6A^6 \sigma^2 + 10A^4 \sigma^4 + 4A^2 \sigma^6 + \sigma^8) + (N_{\text{seq}}' - N_{\text{valid}}) (A_0 A_1)^8} \\
\text{SNR}_{\text{DPT}_2} &= \frac{N_{\text{valid}}^2 (A_0 A_1)^4}{N_{\text{seq}} (2A^2 \sigma^2 + \sigma^4) + (N_{\text{seq}} - N_{\text{valid}}) (A_0 A_1)^4} \\
\text{SNR}_{\text{DFrFT}} &= \frac{N_{\text{valid}}^2 (A_0 A_1)^2}{N_{\text{seq}} \sigma^2 + (N_{\text{seq}} - N_{\text{valid}}) (A_0 A_1)^2},
\end{aligned} \tag{31}$$

respectively. The SNRs will be much lower than 0 dB when $N_{\text{valid}} \ll N_{\text{seq}}$, which leads to an inaccurate estimation result.

Fortunately, N_{valid} can be obtained by (30) with a given a_{1m} , and $N_{\text{valid}} \approx N_{\text{seq}}$ can be ensured by selecting the data length.

Thus, we assume the pilot length is sufficient for estimation, and (31) can be simplified as follows in the high SNR regime:

$$\begin{aligned} \text{SNR}_{\text{DPT}_3} &\approx \frac{N'_{\text{seq}} (A_0 A_1)^8}{6A^6 \sigma^2 + 10A^4 \sigma^4 + 4A^2 \sigma^6 + \sigma^8} \\ &\approx \frac{N'_{\text{seq}} (A_0 A_1)^2}{6\sigma^2}, \\ \text{SNR}_{\text{DPT}_2} &\approx \frac{N_{\text{seq}} (A_0 A_1)^4}{2A^2 \sigma^2 + \sigma^4} \approx \frac{N_{\text{seq}} (A_0 A_1)^2}{2\sigma^2}, \\ \text{SNR}_{\text{DFrFT}} &\approx \frac{N_{\text{seq}} (A_0 A_1)^2}{\sigma^2}. \end{aligned} \quad (32)$$

With the aid of Newton method, the accuracy of \hat{a}_3 , \hat{a}_2 , and \hat{a}_1 based on DPT₃, DPT₂, and DFrFT is improved and given by

$$\begin{aligned} \sigma_{\hat{a}_3}^2 &= \frac{6}{(2\pi)^2 \text{SNR}_{\text{DPT}_3} T_s'^{1/2} (N_{\text{seq}}'^2 - 1)} \\ &\approx \frac{36\sigma^2}{(2\pi)^2 (A_0 A_1)^2 T_s'^{1/2} N_{\text{seq}}' (N_{\text{seq}}'^2 - 1)}, \\ \sigma_{\hat{a}_2}^2 &= \frac{6}{(2\pi)^2 \text{SNR}_{\text{DPT}_2} T_s'^{1/2} (N_{\text{seq}}'^2 - 1)} \\ &\approx \frac{12\sigma^2}{(2\pi)^2 (A_0 A_1)^2 T_s'^{1/2} N_{\text{seq}} (N_{\text{seq}}^2 - 1)}, \\ \sigma_{\hat{a}_1}^2 &= \frac{6}{(2\pi)^2 \text{SNR}_{\text{DFrFT}} T_s'^{1/2} (N_{\text{seq}}'^2 - 1)} \\ &\approx \frac{6\sigma^2}{(2\pi)^2 (A_0 A_1)^2 T_s'^{1/2} N_{\text{seq}} (N_{\text{seq}}^2 - 1)}. \end{aligned} \quad (33)$$

With a given SNR, the accuracy of \hat{a}_3 , \hat{a}_2 , and \hat{a}_1 can be improved by lengthening data or increasing sample interval. Assuming $N'_{\text{seq}} = N_{\text{seq}}$ and $T_s'' \approx T_s'$, the accuracy relationship among \hat{a}_3 , \hat{a}_2 , and \hat{a}_1 is

$$\sigma_{\hat{a}_3}^2 \approx 3\sigma_{\hat{a}_2}^2 \approx 6\sigma_{\hat{a}_1}^2. \quad (34)$$

Note that the accuracy of \hat{a}_2 can be further improved by α -search of DFrFT. Letting the estimation error before α -search and the search range of α -search be $x_{\hat{a}_2}$ and $[\hat{a}_2 - 3\sigma_{\hat{a}_2}, \hat{a}_2 + 3\sigma_{\hat{a}_2}]$, respectively, and assuming that the search times are sufficient, the estimation error after α -search is

$$x'_{\hat{a}_2} = \begin{cases} x_{\hat{a}_2} + 3\sigma_{\hat{a}_2}, & x_{\hat{a}_2} < -3\sigma_{\hat{a}_2}, \\ 0, & x_{\hat{a}_2} \in [-3\sigma_{\hat{a}_2}, 3\sigma_{\hat{a}_2}], \\ x_{\hat{a}_2} - 3\sigma_{\hat{a}_2}, & x_{\hat{a}_2} > 3\sigma_{\hat{a}_2}. \end{cases} \quad (35)$$

The expectation and variance of $x'_{\hat{a}_2}$ are presented in the following equation:

$$\begin{aligned} E(x'_{\hat{a}_2}) &= \int_{-\infty}^{-3\sigma_{\hat{a}_2}} (x_{\hat{a}_2} + 3\sigma_{\hat{a}_2}) \cdot p(x_{\hat{a}_2}) dx_{\hat{a}_2} + \int_{-3\sigma_{\hat{a}_2}}^{3\sigma_{\hat{a}_2}} 0 \\ &\quad \cdot p(x_{\hat{a}_2}) dx_{\hat{a}_2} + \int_{3\sigma_{\hat{a}_2}}^{+\infty} (x_{\hat{a}_2} - 3\sigma_{\hat{a}_2}) \\ &\quad \cdot p(x_{\hat{a}_2}) dx_{\hat{a}_2} = 0, \\ \text{var}(x'_{\hat{a}_2}) &= \int_{-\infty}^{-3\sigma_{\hat{a}_2}} (x_{\hat{a}_2} + 3\sigma_{\hat{a}_2})^2 \cdot p(x_{\hat{a}_2}) dx_{\hat{a}_2} \\ &\quad + \int_{-3\sigma_{\hat{a}_2}}^{3\sigma_{\hat{a}_2}} 0 \cdot p(x_{\hat{a}_2}) dx_{\hat{a}_2} + \int_{3\sigma_{\hat{a}_2}}^{+\infty} (x_{\hat{a}_2} - 3\sigma_{\hat{a}_2})^2 \\ &\quad \cdot p(x_{\hat{a}_2}) dx_{\hat{a}_2} = 2 \left(\int_{-\infty}^{-3\sigma_{\hat{a}_2}} x_{\hat{a}_2}^2 \cdot p(x_{\hat{a}_2}) dx_{\hat{a}_2} \right. \\ &\quad \left. + 6\sigma_{\hat{a}_2} \int_{-\infty}^{-3\sigma_{\hat{a}_2}} x_{\hat{a}_2} \cdot p(x_{\hat{a}_2}) dx_{\hat{a}_2} \right. \\ &\quad \left. + 9\sigma_{\hat{a}_2}^2 \int_{-\infty}^{-3\sigma_{\hat{a}_2}} p(x_{\hat{a}_2}) dx_{\hat{a}_2} \right) = 10 \left(1 \right. \\ &\quad \left. - \text{erf}\left(\frac{3}{\sqrt{2}}\right) \right) \sigma_{\hat{a}_2}^2 \approx 0.027\sigma_{\hat{a}_2}^2, \end{aligned} \quad (36)$$

with $p(x_{\hat{a}_2}) = (1/\sqrt{2\pi\sigma_{\hat{a}_2}^2})e^{-x_{\hat{a}_2}^2/2\sigma_{\hat{a}_2}^2}$ and $\text{erf}(x) = (2/\sqrt{\pi}) \int_0^x e^{-t^2} dt$. If the search range becomes to $[-2\sigma_{\hat{a}_2}, 2\sigma_{\hat{a}_2}]$, the variance turns into $0.35\sigma_{\hat{a}_2}^2$ accordingly.

Then we discuss the accuracy of \hat{a}_0 , which is derived from DFrFT due to (21). In the high SNR regime, \hat{a}_0 follows a zero-mean Gaussian distribution with variance

$$\sigma_{\hat{a}_0}^2 = \frac{\sigma^2}{(2\pi)^2 N_{\text{seq}} (A_0 A_1)^2 \text{sinc}^2(\Delta f)}, \quad (37)$$

where Δf denotes the offset between the DFT result and the true value.

Finally, the performance of SFFT is discussed as follows. Letting $\hat{X}_{-\Gamma}(k) = \{\hat{X}(k) \mid k \notin \Gamma\}$, the estimation error probability based on SFFT is [22]

$$\begin{aligned} P \left[\left| \hat{X}(k) - X(k) \right|^2 \geq \frac{\varepsilon}{l} \left\| \hat{X}_{-\Gamma}(k) \right\|_2^2 + 3\delta^2 \left\| \hat{X}(k) \right\|_1^2 \right] \\ < O \left(\frac{l}{\varepsilon B} \right). \end{aligned} \quad (38)$$

SFFT performs similar to FFT with a sufficient small error probability if $l \ll B$. Consequently, the accuracies of \hat{a}_3 , \hat{a}_2 , \hat{a}_1 , and \hat{a}_0 based on SFFT are approximately equal to those based on FFT, which demonstrates the agreement between the proposed sparse algorithm and nonsparse algorithm.

TABLE 1: The algorithm complexity.

Approach	Multiplies	Adds
Correlation	$2f_s L_h T_{syb}$	$2(f_s L_h T_{syb} - 1)$
DP2	N_{seq}	0
DFT		
FFT	$\frac{M}{2} \log_2 M$	$M \log_2 M$
SFFT	$\left(\omega + \frac{B}{2} \log_2 B + \frac{LM}{B}\right) L$	$\left(B \log_2 B + \frac{LM}{B}\right) L$
DFFrFT		
DFFrFT	$\left(M + \frac{M}{2} \log_2 M\right) \lambda_\alpha$	$(M \log_2 M) \lambda_\alpha$
SDFrFT	$\left(M + \left(\omega + \frac{B}{2} \log_2 B + \frac{LM}{B}\right) L\right) \lambda_\alpha$	$\left(B \log_2 B + \frac{LM}{B}\right) L \lambda_\alpha$
Rife & Newton methods	$(3N_{seq} + 1) \lambda_{itrt} + 2$	$(2N_{seq} - 1) \lambda_{itrt} + 2$

4.2. Algorithm Complexity Analysis. The algorithm complexity is shown in Table 1, where λ_{itrt} and λ_α denote the iteration times of Newton method and the α -search times, respectively. Referring to the values of ω , B , and L in Algorithm 1, the complexities of SFFT and FFT can be expressed as $O(\log_2 M \sqrt{M l \log_2 M})$ and $O(M \log_2 M)$, respectively. In the large data size regime, the complexity of FFT is about M times more than that of SFFT.

In the condition of $N_{seq} = M$, considering DP₂ and Rife and Newton methods are employed three times, respectively, and SDFrFT and SFFT are employed twice, respectively, the total numbers of multiplications, C_{mul} , and additions, C_{add} , in the proposed sparse algorithm are

$$C_{mul} \approx (9\lambda_{itrt} + 2\lambda_\alpha) N_{seq} + 2\lambda_\alpha \log_2 \left(N_{seq} \sqrt{N_{seq} l \log_2 N_{seq}} \right), \quad (39)$$

$$C_{add} \approx 6\lambda_{itrt} N_{seq} + 2\lambda_\alpha \log_2 \left(N_{seq} \sqrt{N_{seq} l \log_2 N_{seq}} \right).$$

Similarly, the total numbers of multiplications, \dot{C}_{mul} , and additions, \dot{C}_{add} , in the nonsparse algorithm are

$$\dot{C}_{mul} \approx 9\lambda_{itrt} N_{seq} + \lambda_\alpha N_{seq} \log_2 N_{seq}, \quad (40)$$

$$\dot{C}_{add} \approx 6\lambda_{itrt} N_{seq} + 2\lambda_\alpha N_{seq} \log_2 N_{seq}.$$

Figure 7 presents the complexity comparison of the proposed sparse algorithm with the nonsparse algorithm and the DDLL algorithm. The results indicate that the proposed algorithm has a much lower complexity than both the nonsparse algorithm and the DDLL algorithm.

5. Implementation Issues

In this section, we discuss the implementation issues of the proposed algorithm including the module reuse analysis and the clock rate analysis.

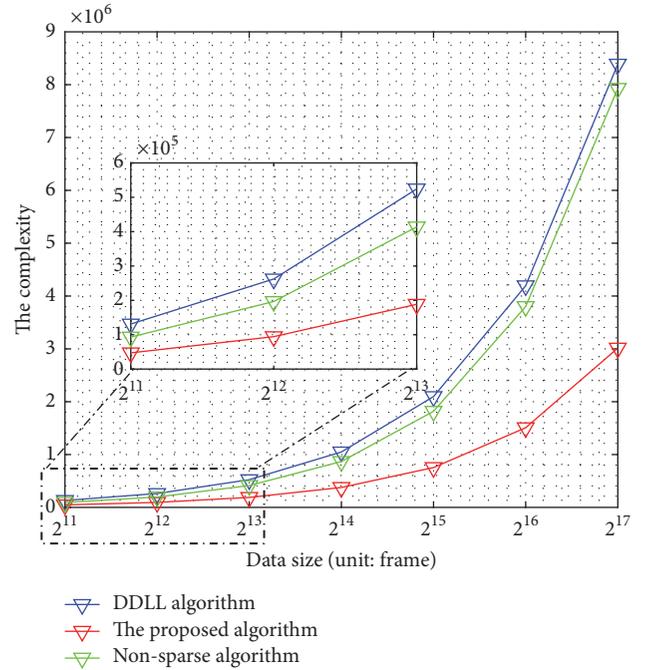


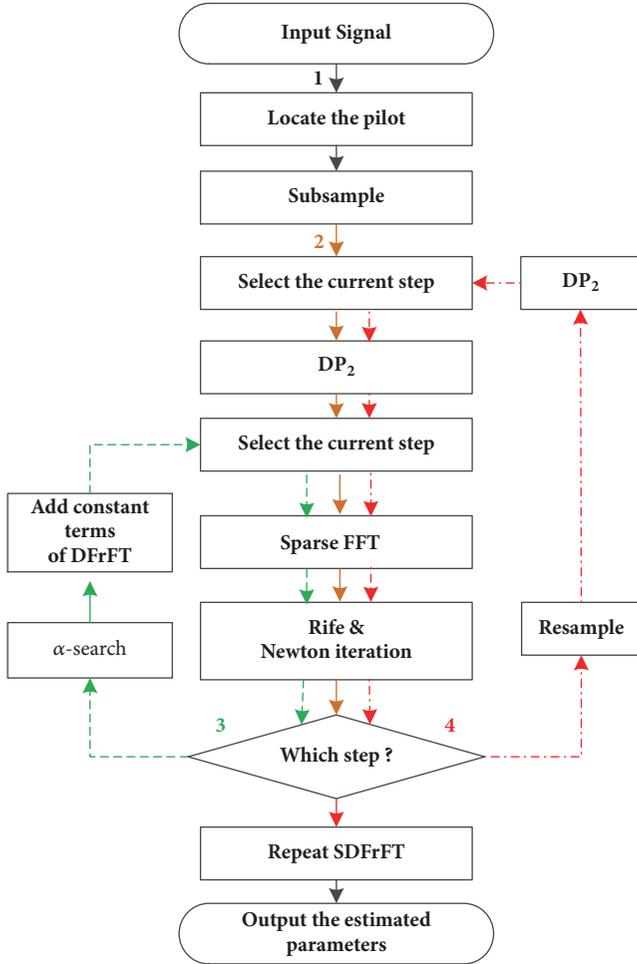
FIGURE 7: The comparison in terms of complexity.

5.1. Module Reuse Analysis. In NTN platforms, with the requirement of low power consumption and low cost, the chip size is strictly limited, which means that the reuse of system module is necessary.

Figure 8 presents the implementation procedure of the proposed algorithm. According to Section 3, it mainly contains four steps, among which some modules can be reused. First, DP₃ can be conducted by twice DP₂ due to (24). Thus, SDPT₃ and SDPT₂ share a DP₂ module. Second, SDPT and SDFrFT share an SFFT module. Third, SDPT₂, SDPT₃, and SDFrFT share a “Rife and Newton methods” module. Consequently, a correlation module, a DP₂ module, an SFFT module, and a “Rife and Newton methods” module are sufficient for the whole procedure. We just need $f_s L_h T_{syb}$ real

TABLE 2: The comparison between DDLL and the proposed algorithm.

Items	The DDLL Algorithm	The proposed Algorithm	Ratio
Process clock rate	156.25 MHz	610.35 kHz	256:1
Degree of parallelism	64	1	64:1



Note:
 — 1: the first step. - - - 3: the third step.
 — 2: the second step. - · - · 4: the fourth step.

FIGURE 8: The implementation of the proposed algorithm.

multipliers, about $(B \log_2 B + lM/B + 2N_{seq} + f_s L_h T_{syb})$ adders, and about $(\omega + (B/2) \log_2 B + (l/B + 1)M + 4N_{seq})$ complex multipliers in total, which is easy to implement on chip, e.g., on the Field Programmable Gate Array (FPGA).

5.2. Clock Rate Analysis. The clock rate is a crucial parameter in high speed system. A 5 Gbps communication system requires a clock rate of $f_s \geq 10$ GHz, which is difficult to implement on FPGA as the maximum rate of FPGA is on the order of hundreds of MHz. To achieve this target clock rate, the parallel processing is required in the conventional DDLL

TABLE 3: The critical parameters.

Parameter	Value
Modulation	OOK
Demodulation	Direct detection
Data rate	5 Gbps
SNR	14 dB
The maximum velocity	7 km/s
The maximum acceleration	800 m/s ²
The maximum jerk	60 m/s ³
L_{frm}	8192 bit
L_{syn}	32 bit
L_p	49 bit

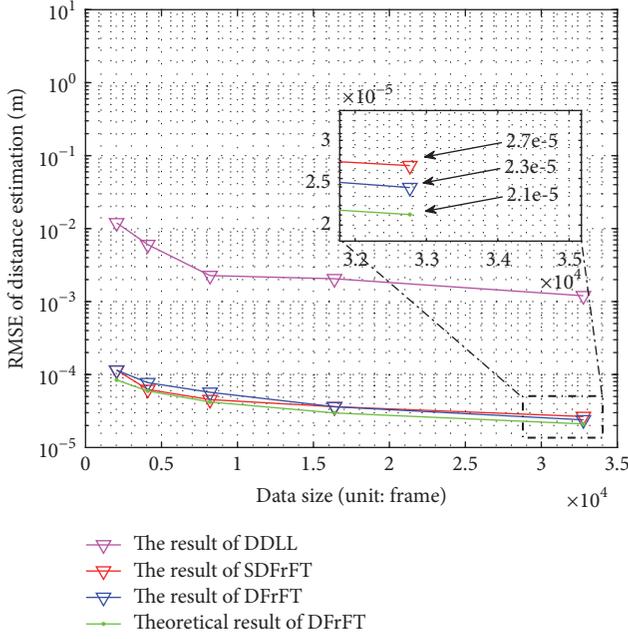
algorithm, which would lead to high resource consumption. In comparison, the clock rate and the degree of parallelism can be decreased in the proposed sparse algorithm thanks to the subsampling, and the power consumption is accordingly decreased to a great extent.

The comparison between the DDLL algorithm and the proposed algorithm in terms of clock rate and degree of parallelism are shown in Table 2 with $L_{frm} = 8192$ bit, $\beta = 1$, $f_s = 10$ GHz, and $r_{smp} = \beta L_{frm} T_{syb} / T_s = 16384$. The numerical results demonstrate the superiority of the proposed algorithm over the DDLL algorithm in terms of power consumption. From the above discussions, the proposed algorithm can be applied to the spaceborne platform of 5G NTN.

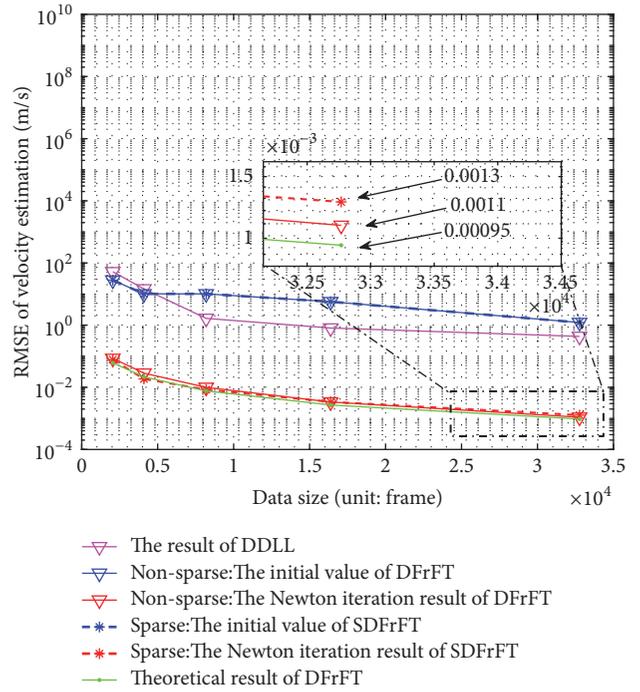
6. Simulation Results

In this section, we evaluate the performance of the proposed sparse algorithm in comparison with the nonsparse algorithm and the DDLL algorithm. The main parameters are listed in Table 3. In the simulation, we first verify the root-mean-square error (RMSE) of these three algorithms versus data size, N_{seq} . As shown in Figure 9, the theoretical results derived from (33), (36), and (37) are presented by the green lines in the figures. It is observed that the simulation results match well with theoretical results.

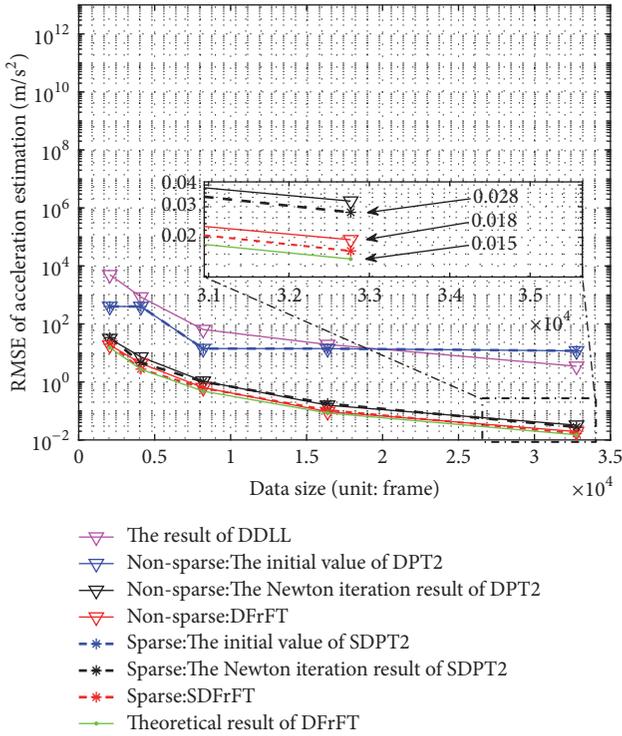
Specifically, Figure 9(a) plots the RMSE of distance estimation (\hat{a}_0) versus N_{seq} . Although there exists a small gap between the results of DFrFT and SDFrFT, the proposed sparse algorithm can still accurately estimate a_0 . Moreover, it outperforms the DDLL algorithm, especially in the small data size regime. The reason is that DDLL needs long convergence time to achieve high-precision estimation in high dynamic environment. Although the convergence rate can be



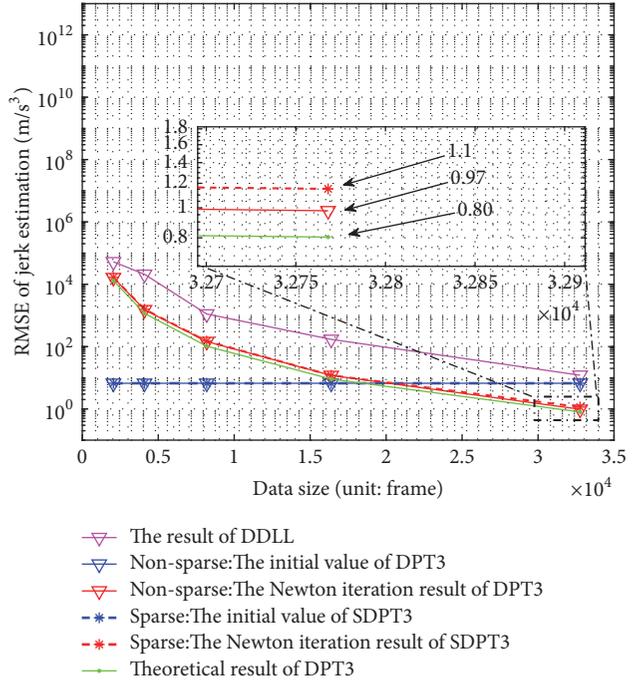
(a) The RMSE of distance estimation



(b) The RMSE of velocity estimation



(c) The RMSE of acceleration estimation



(d) The RMSE of jerk estimation

FIGURE 9: The comparison of the three algorithms versus N_{seq} .

improved by increasing the loop bandwidth, it results in a high steady-state error and violates the accuracy requirement.

Figure 9(b) plots the RMSE of velocity estimation (\hat{a}_1) versus N_{seq} . SDFrFT exhibits a similar performance to DFrFT

on estimating a_1 , which is similar to that on estimating a_0 . Furthermore, with the aid of Newton method, SDFrFT outperforms the DDLL algorithm, especially in the small data size regime. From the result, a small L_p is sufficient for

estimating a_1 , which indicates that the proposed algorithm can be implemented without enormous channel cost.

Figure 9(c) plots the RMSE of acceleration estimation (\hat{a}_2) versus N_{seq} . As shown in the figure, the accuracy of DPT₂-based result is improved by Newton method and further improved by the α -search of DFrFT, which can be explained by the analysis in Section 4. The improved estimated result has a better accuracy than the DDLL-based result. In addition, in the small data size regime, there exists a larger gap between the results of the proposed algorithm and the DDLL algorithm in Figure 9(c) than that in Figure 9(b). This is because the transformation-domain algorithm is designed for estimating the acceleration, and DDLL responds more slowly in tracking the acceleration compared with tracking the velocity.

Figure 9(d) plots the RMSE of jerk estimation (\hat{a}_3) versus N_{seq} . The comparison among DPT₃, SDPT₃, and DDLL indicates that there are a huge gap and a minor gap between the results of the proposed algorithm and the DDLL algorithm and between the results of the proposed sparse algorithm and the nonsparse algorithm, respectively. Moreover, in the small data size regime, in comparison to the acceleration estimation results in Figure 9(c), there exists a larger gap between the results of the proposed algorithm and the DDLL algorithm in terms of the jerk estimation. The reason is that SDPT₃ is designed for estimating the jerk, and DDLL responds more slowly in tracking the jerk than the acceleration, which presents the advantage of the proposed algorithm in high dynamic environment.

Next, we focus on the comparison between the sparse algorithm and nonsparse algorithm versus parameter β in the formula $r_{smp} = \beta L_{frm} T_{sybl} / T_s$ with $N_{seq} = 2048$. The simulation results are plotted in Figure 10 and match well with the theoretical results.

Specifically, Figure 10(a) presents the RMSE of distance estimation (\hat{a}_0) versus β . There exists no pronounced difference between the results of SDFrFT and of DFrFT, which agrees with the results in Figure 9(a). However, the variation tendency of the results in Figure 10(a) is different from that in Figure 9(a). The RMSE decreases with the increase of N_{seq} in Figure 9(a) and remains constant no matter the value of β in Figure 10(a). It can be explained by (37) that the accuracy of \hat{a}_0 is related to data size and is independent of the sample interval.

Figures 10(b) and 10(c) plot the RMSEs of velocity estimation (\hat{a}_1) and acceleration estimation (\hat{a}_2) versus β , respectively. The results indicate that the performance is almost the same no matter the sparse method is used or not, which verifies the rationality of the sparse algorithm. Compared with lengthening data, increasing sample interval only has a slighter effect on estimating a_1 and a_2 in the condition of the same process duration. It can be explained by (33) that lengthening data contributes more to the accuracy than increasing sample interval. Fortunately, the difference between these two cases is so small that an

accurate estimation still can be achieved by increasing the interval. As increasing sample interval improves the estimation accuracy without increasing computational burden, we prefer it rather than lengthening data for the sake of power consumption.

From the results in Figures 9(d) and 10(d), lengthening data performs better than increasing sample interval in the same process duration, which is similar to the comparison results of Figures 9(b), 9(c), 10(b), and 10(c). The gap can be filled up by slightly increasing the data size, e.g., as shown in Figure 11 with $N_{seq} = 4096$, whose results indicate that the configuration of $N_{seq} = 4096$ and $\beta = 16$ exhibits a prior performance to the configuration of $N_{seq} = 32768$ and $\beta = 1$ in Figure 9(d), and a much better performance than that of $N_{seq} = 2048$ and $\beta = 16$ in Figure 10(d).

From the above discussion, as increasing sample interval improves the estimation accuracy without increasing computation resource, it is a better choice than lengthening data in resource constrained platforms.

7. Experiment Results

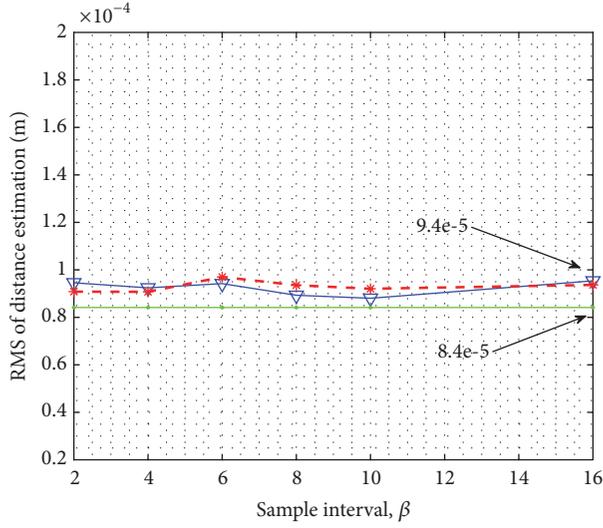
Referring to the parameters listed in Table 3, we have built a hardware platform for the experiment to verify the implementability of the proposed algorithm. The topology of hardware platform is given in Figure 12. The laser signal is received by telescope and is transformed to an electrical signal in the optical receiver. After that, the electrical signal is transmitted to channel simulator, which is employed to tune the transmission delay a_0 and delay rate a_1 with the control of PC, via the LAN extension for instrumentation (LXI) protocol. The output of channel simulator is sampled by an analog-digital converter (ADC). Then FPGA utilizes the samples to implement the proposed temporal synchronization algorithm according to Figure 8. The synchronization results are reported to PC, via the peripheral component interconnect express (PCI-E) protocol.

We tune the transmission delay and delay rate by channel simulator and estimate them by the proposed algorithm with $M = N_{seq} = 2048$ and $\beta = 2$. The experiment is repeated 50 times for each configuration of a_0 and a_1 , and the statistical results are presented in Table 4, where the RMSEs of transmission delay and delay rate are on the orders of 0.01 ns and 0.3 ns/s, respectively. Due to the performance loss caused by the nonideal hardware and the nonideal channel, the experimental result performs inferior to the simulation result. Fortunately, the accuracy of experimental result still meets the demand of 5G NTN, and the performance can be further improved by lengthening the process duration.

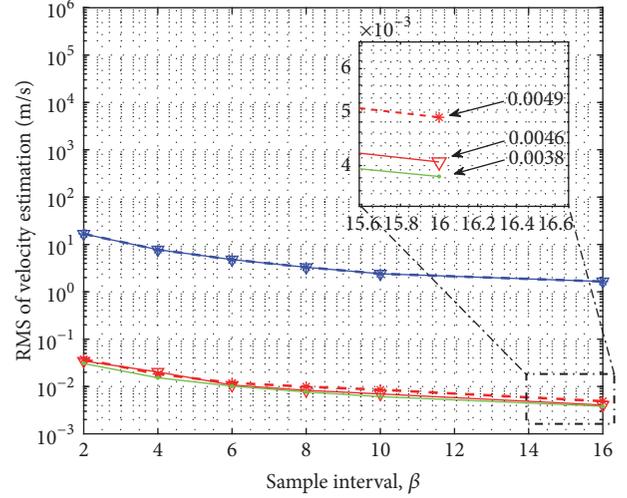
Next, we repeat the proposed temporal synchronization algorithm once per 50 milliseconds to continuously estimate the transmission delay, and the result is depicted in Figure 13. The result in Figure 13 indicates that the proposed algorithm can be implemented to continuously estimate the temporal parameters in real time, and the accuracy achieves about 0.01

TABLE 4: The experiment results of temporal synchronization.

True value of a_0 (ns)	True value of a_1 (ns/s)	RMSE of \hat{a}_0 (ns)	RMSE of \hat{a}_1 (ns/s)
1	100	0.0103	0.264
1	1000	0.0100	0.311
10	100	0.0168	0.192
10	1000	0.0270	0.220
100	100	0.0138	0.268
100	1000	0.0133	0.306
1000	100	0.0185	0.212
1000	1000	0.0265	0.109



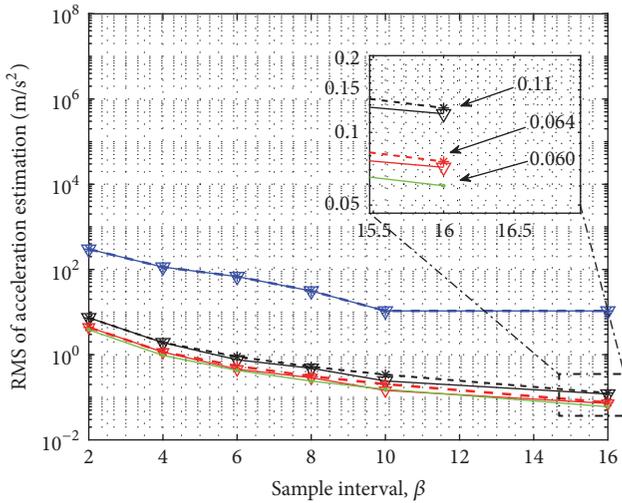
- ▽ The result of DFrFT
- * The result of SDFrFT
- Theoretical result of DFrFT



- ▽ The initial value of DFrFT
- ▽ The Newton iteration result of DFrFT
- * The initial value of SDFrFT
- * The Newton iteration result of SDFrFT
- Theoretical result of DFrFT

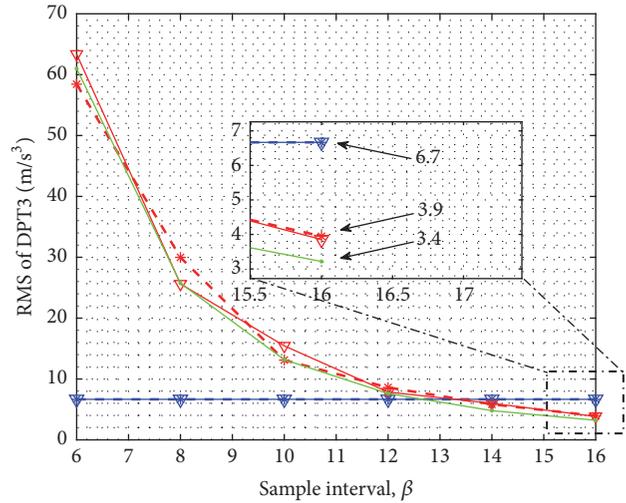
(a) The RMSE of distance estimation

(b) The RMSE of velocity estimation



- ▽ The initial value of DPT2
- ▽ The Newton iteration result of DPT2
- ▽ The result of DFrFT
- * The initial value of SDPT2
- * The Newton iteration result of SDPT2
- * The result of SDFrFT
- Theoretical result of DFrFT

(c) The RMSE of acceleration estimation



- ▽ The initial value of DPT3
- ▽ The Newton iteration result of DPT3
- * The initial value of SDPT3
- * The Newton iteration result of SDPT3
- Theoretical result of DPT3

(d) The RMSE of jerk estimation

FIGURE 10: The comparison of the sparse algorithm and nonsparse algorithm versus β with $N_{seq} = 2048$.

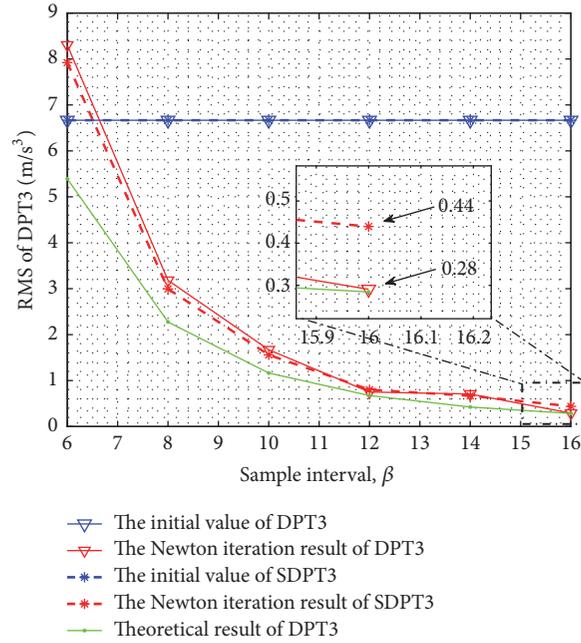


FIGURE 11: The jerk estimation versus β with $N_{seq} = 4096$.

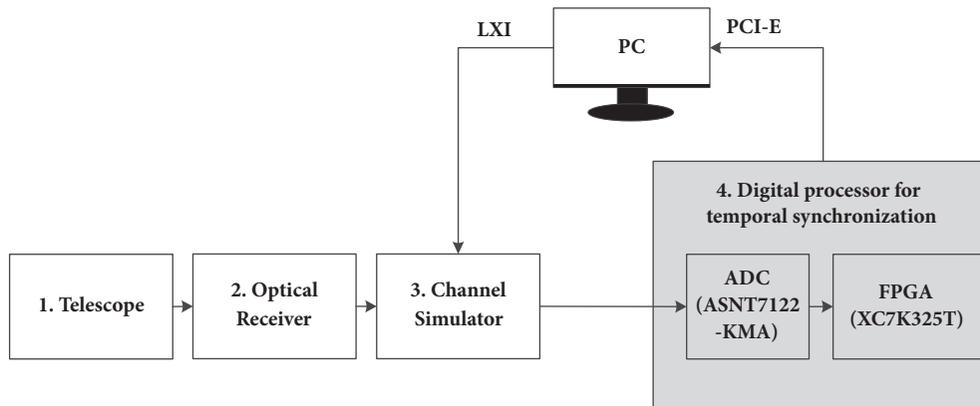


FIGURE 12: The topology of hardware platform.

ns in experimental environment. As the employed channel simulator cannot simulate the acceleration and jerk currently, the performance on estimating a_2 and a_3 will be verified in follow-up experiments.

8. Conclusions

In this paper, we have developed a temporal synchronization algorithm based on the sparse pilot and the sparse transform method in the scenario with high Doppler and high data rate. The performance of the proposed algorithm has been analyzed in comparison with the conventional DLL algorithm and the nonsparse algorithm. The analytical and simulation results demonstrate that the proposed algorithm performs

best in terms of a good accuracy and the lowest complexity. In addition, we have implemented the proposed algorithm, which confirms its implementability in resource constrained platforms, e.g., the IMDD-based laser communications for feeder links in 5G NTN platforms.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

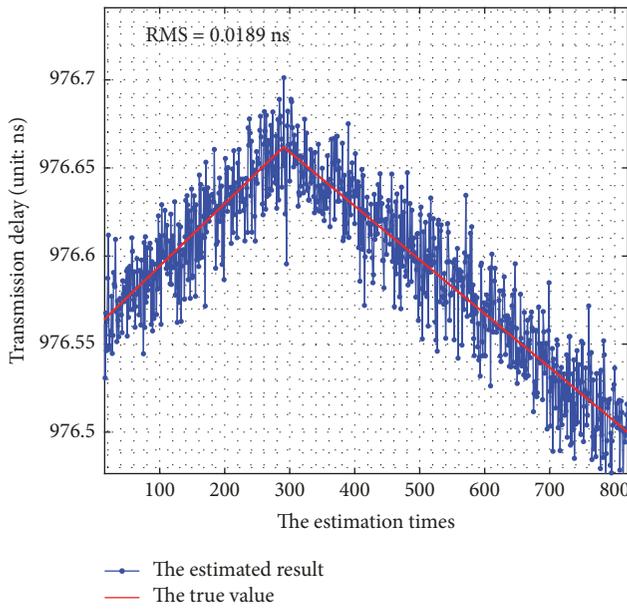


FIGURE 13: The experiment result of continuous estimation.

Acknowledgments

This work was funded by Research on Space-Oriented Terahertz Wideband Communication Theory and Technology [6161001093].

References

- [1] B. Soret, A. De Domenico, S. Bazzi, N. H. Mahmood, and K. I. Pedersen, "Interference Coordination for 5G New Radio," *IEEE Wireless Communications Magazine*, pp. 1–7, 2017.
- [2] J. An, K. Yang, J. Wu, N. Ye, S. Guo, and Z. Liao, "Achieving Sustainable Ultra-Dense Heterogeneous Networks for 5G," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 84–90, 2017.
- [3] D. B. Mckenna and B. J. Cox, "Doppler insensitive non-terrestrial digital cellular communications network," *Patent US6377802*, vol. 23, 2002.
- [4] B. A. Lauer, K. Wang, T. Lamarca et al., "Systems and methods for facilitating communications originating from a non-terrestrial network," *Patent US9503956*, Nov 2016.
- [5] B. A. Lauer, K. Wang, T. Lamarca et al., "Systems and methods for facilitating communications destined for a non-terrestrial network," *Patent WO/2015/184339*, Article ID 184339, Dec 2015.
- [6] S. Poulernard, M. Crosnier, and A. Rissons, "Ground segment design for broadband geostationary satellite with optical feeder link," *IEEE/OSA Journal of Optical Communications & Networking*, vol. 7, no. 4, pp. 325–336, Apr 2015.
- [7] B. Roy, S. Poulernard, S. Dimitrov et al., "Optical feeder links for high throughput satellites," in *Proceedings of the IEEE International Conference on Space Optical Systems and Applications (ICSOS '15)*, pp. 1–6, New Orleans, LA, USA, October 2015.
- [8] N. Perlot, T. Dreischer, C. M. Weinert et al., "Optical GEO feeder link design," in *Proceedings of the FutureNetw*, pp. 1–8, Berlin, Germany, 2013.
- [9] H. Someya, I. Oowada, H. Okumura, T. Kida, and A. Uchida, "Synchronization of bandwidth-enhanced chaos in semiconductor lasers with optical feedback and injection," *Optics Express*, vol. 17, no. 22, pp. 19536–19543, 2009.
- [10] D. O. Caplan, "Laser communication transmitter and receiver design," *Journal of Optical Fiber Communications Reports*, vol. 4, no. 4-5, pp. 225–362, Sept 2007.
- [11] J. Li, H. Ruan, and F. Liu, "The design and analysis of PD radar aid by GPS/INS ultra-tightly coupled navigation," in *Proceedings of the IET International Radar Conference 2013*, China, April 2013.
- [12] Y. Li, Z. Tao, and Z. X. Niu, "Ultra-wide bandwidth Communication signal tracking Algorithm based on delay-locked Loop," *Transactions of Beijing Institute of Technology*, vol. 26, no. 11, pp. 1019–1021, Nov 2006.
- [13] R. Alhakim, K. Raoof, and E. Simeu, "Design of tracking loop with dirty templates for UWB communication systems," *Signal Image & Video Processing*, vol. 8, no. 3, pp. 461–477, Mar 2014.
- [14] Z. Y. Zhan, J. S. Ding, W. Y. Wu, and J. G. Hou, "A new tracking method for high dynamic DS-SS signal," in *Proceedings of the IET International Radar Conference '15*, pp. 1–5, Hangzhou, China, 2015.
- [15] C. Lin, B. Shao, and J. Zhang, "A high data rate parallel demodulator suited to FPGA implementation," in *Proceedings of the International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS '10)*, pp. 1–4, Chengdu, China, December 2010.
- [16] D. Schmidt and B. Lankl, "Parallel architecture of an all digital timing recovery scheme for high speed receivers," in *Proceedings of the CSNDSP Newcastle upon Tyne*, pp. 31–34, UK, 2010.
- [17] C. Lin, J. Zhang, and B. Shao, "A High Speed Parallel Timing Recovery Algorithm and Its FPGA Implementation," in *Proceedings of the 2nd International Symposium on Intelligence Information Processing and Trusted Computing (IPTC '11)*, pp. 63–66, Wuhan, China, October 2011.
- [18] W. C. Du, X. Q. Gao, and G. H. Wang, "Using FRFT to estimate target radial acceleration," in *Proceedings of the International Conference on Wavelet Analysis and Pattern Recognition '07*, pp. 442–447, Beijing, China, November 2007.
- [19] R. Brcich and A. Zoubir, "The use of the DPT in passive acoustic aircraft flight parameter estimation," in *Proceedings of the IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications*, pp. 819–822, Brisbane, Qld, Australia, 1997.
- [20] F. Lu, H. Wang, X. Liu et al., "Time-frequency characteristics of PSWF with Wigner-Ville Distributions," in *Proceedings of the ICSIP*, pp. 568–582, Beijing, China, 2016.
- [21] F. Pignol, F. Colone, and T. Martelli, "Lagrange polynomial interpolation based Keystone Transform for passive radar," *IEEE Transactions on Aerospace & Electronic Systems*, vol. 99, 2017.
- [22] H. Hassanieh, P. Indyk, D. Katabi, and E. Price, "Simple and practical algorithm for sparse Fourier transform," in *Proceedings of the Acm-Siam Symposium on Discrete Algorithms*, pp. 1183–1194, San Francisco, CA, USA.
- [23] A. C. Gilbert, P. Indyk, M. Iwen, and L. Schmidt, "Recent developments in the sparse fourier transform: a compressed fourier transform for big data," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 91–100, 2014.
- [24] J. Schumacher, *High Performance Sparse Fast Fourier Transform*, Dept. Computer Science, ETH Zurich, Zurich, Switzerland, 2013.

- [25] S. Liu, T. Shan, R. Tao et al., "Sparse discrete fractional fourier transform and its applications," *IEEE Transactions on Signal Processing*, vol. 62, no. 24, pp. 6582–6595, 2014.
- [26] A. Chaaban and M. S. Alouini, "Optical intensity modulation direct detection versus heterodyne detection: A high-SNR capacity comparison," in *Proceedings of the 5th International Conference on Communications and Networking (COMNET '15)*, pp. 1–5, Tunis, Tunisia, November 2015.
- [27] G. N. Liu, L. Zhang, T. Zuo, Q. Zhang, J. Zhou, and E. Zhou, "IM/DD Transmission Techniques for Emerging 5G Fronthaul, DCI and Metro Applications," in *Proceedings of the Optical Fiber Communication Conference*, pp. 1–3, Los Angeles, California, 2017.
- [28] M. A. Khalighi and M. Uysal, "Survey on Free Space Optical Communication: A Communication Theory Perspective," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 2231–2258, 2014.
- [29] S. Arnon, J. Barry, G. Karagiannidis, R. Schober, and M. Uysal, *Advanced Optical Wireless Communication Systems*, Cambridge University Press, New York, NY, USA, 1st edition, 2012.
- [30] M. Y. Hong, "A fast sinusoidal signal analysis technique for the determination of complex frequencies," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), '02*, pp. 1469–1472, Orlando, FL, USA, 2002.
- [31] T. J. Abatzoglou, "A Fast Maximum Likelihood Algorithm for Frequency Estimation of a Sinusoid Based on Newton's Method," *IEEE Transactions on Acoustics Speech & Signal Processing*, vol. 33, no. 1, pp. 77–89, Mar 1985.
- [32] S. Peleg, B. Porat, and B. Friedlander, "The discrete polynomial transform (DPT), its properties and applications," in *Proceedings of the Conference Record of the Twenty-Fifth Asilomar Conference on Signals, Systems & Computers*, pp. 116–120, Pacific Grove, CA, USA, 2002.

