

# Low Complexity Methods for Joint Detection and Synchronization of TDMA Bursts

Haotian Zhai and Bernd-Peter Paris  
*Department of Electrical and Computer Engineering*  
*George Mason University*  
Fairfax, VA 22030  
{hzhai,pparis}@gmu.edu

**Abstract**—This paper proposes a data-aided joint detection and synchronization algorithm for TDMA bursts. A sequential detection algorithm based on the Generalized Likelihood Ratio Test (GLRT) is used to detect the embedded preamble signal in received data stream. Carrier synchronization is attempted for each sample instant during sequential detection and resulting phase and frequency estimates are used by the GLRT. To make this algorithm implemented in software-defined radio (SDR), a low complexity carrier estimation algorithm with good estimation accuracy is proposed. Then, a refinement of the carrier estimate is computed when the preamble has been detected. It is shown that this estimate approaches the Cramer-Rao bound even at low SNR. The complete joint detection and estimation procedure is validated through simulations and SDR.

**Index Terms**—Joint detection and estimation, GLRT, low computational complexity, Cramer-Rao vector bound, Software-defined radio.

## I. INTRODUCTION

In digital communication systems, information is commonly transmitted in time-multiplexed bursts. Examples include time-slotted random access systems. Each active user transmits information to the receiver in the same frequency band and in non-overlapping time intervals [1]. A fundamental prerequisite for successful coherent demodulation is that the receiver can detect the beginning of the data stream and estimate accurately the phase and frequency offset of the carrier. It is worth emphasizing that time and carrier synchronization are coupled problems, especially at low SNR: coherent methods for detecting the signal require accurate frequency and phase estimates while data-aided frequency and phase estimation requires that the location of the training sequence is available. Thus, joint signal detection and carrier synchronization algorithms play a vital role in any communication system; both accuracy and computational complexity of the algorithms must be considered.

Clearly, the signal acquisition problem has been considered widely. Many carrier estimation algorithms have been proposed for burst-mode transmission of digital data in the last 30 years. In the late 90's, Morelli and Mengali [2] presented a tutorial review of the field comparing such characteristics as estimation accuracy, range, and computational complexity of available techniques. The work by [3]–[8] is most closely related to results in this paper. For signal detection and time synchronization, algorithms have been developed for radar and

radio systems with burst-mode transmission [9]–[14]. In these works, the detection problem in the presence of unknown parameters (e.g., delay or Doppler shift) is usually solved by a Generalized Likelihood Ratio Test (GLRT).

Modern communication systems differ from those considered when the majority of synchronization algorithms were developed at least two key aspects. It is common now that communication systems operate at low SNR levels near 0 dB. At low SNR, it is critical to address detection and carrier acquisition jointly as non-coherent detection, assumed by the work cited in [2], is unreliable. Also, greatly increased data rates require that any synchronization algorithm is computationally efficient. Signal acquisition is the first step in the receiver's processing chain and must be performed in real time at the sample rate.

With these considerations in mind, this paper seeks to make the following contributions:

- We propose a low-complexity joint carrier estimation and detection algorithm; the per-sample computational complexity is linear in the length of the reference sequence.
- We offer new closed-form expressions for the distribution of the frequency estimate in the proposed algorithm; these are believed to be useful beyond this paper.
- The signal acquisition chain, including delay estimation and carrier synchronization, is analyzed and implemented on software-defined radio (SDR).

This paper first specifies our assumptions about the signal model. Then, the first step of the signal acquisition in this paper, i.e., the time synchronization problem is addressed. Our algorithm is a sequential GLRT that relies on a joint frequency and phase estimate to facilitate coherent detection. Next, the focus shifts to estimating the carrier's phase and frequency offset. A family of low-complexity and high-accuracy estimation algorithms is proposed. Performance analysis of the complete system is provided through simulation. At the end of the paper, we perform the complete signal acquisition chain and the proposed joint detection and estimation algorithms on software-defined radio (SDR).

## II. SIGNAL MODEL

The transmitted signal burst is assumed to include a reference signal that is known to the receiver. Often such a reference sequence is prepended to the payload and is referred to as a

preamble. The problem addressed in this paper is to accurately estimate the start time of the preamble and to estimate carrier phase and frequency offset from this preamble. Hence, the payload portion of the burst is not further considered.

The baseband-equivalent reference signal  $s(t)$  is digitally modulated as

$$s(t) = \sum_{i=0}^{L_0-1} c_i g(t - iT), \quad (1)$$

where  $T$  denotes the symbol period and  $g(t)$  provides pulse shaping.  $\{c_i\}_{i=0}^{L_0-1}$  is the known symbol sequence between transmitter and receiver, where  $L_0$  denotes the number of symbols. It is common to assume the symbols  $c_i$  have good autocorrelation properties to render coherent processing effective.

At the receiver side, the received baseband equivalent signal stream are given by

$$r(t) = s(t - \tau) \cdot A e^{j\phi} e^{j2\pi f_d t} + w(t). \quad (2)$$

where  $\tau$  denotes the delay (the start time) of received reference signal in the stream.  $A$ ,  $\phi$ ,  $f_d$  are the carrier amplitude, phase, and frequency offset, respectively. These are the parameters to be estimated for signal acquisition in this paper. The complex, additive white Gaussian noise is denoted  $w(t)$ .

Our algorithms are discussed in discrete-time; sampling the received signal  $r(t)$  at a rate of  $M$  samples per symbol, i.e., the sampling frequency  $f_s = \frac{1}{T_s} = \frac{M}{T}$ , yields

$$r_n = s_{n-p}^{(\Delta p)} A e^{j\phi} e^{j2\pi \delta n} + w_n, \quad (3)$$

where, the continuous-time delay  $\tau$  is decomposed into  $\tau = pT_s + \Delta p$  with  $pT_s$  integer multiples of sample period and the fractional delay  $\Delta p$  satisfying  $-T_s/2 < \Delta p \leq T_s/2$ . Thus,  $\Delta p$  is negatively correlated with the sampling frequency. Moreover, the sampled received reference sequence, including the effect of fractional delay, is

$$s_n^{(\Delta p)} = \sum_{i=0}^{L_0-1} c_i g(nT_s - iT - \Delta p) \quad \text{for } n = 0, \dots, N-1, \quad (4)$$

where  $N = ML_0$  is the number of samples in the preamble. From (4), it can be seen the value of  $\Delta p$  will affect (degrade) the performance of both detection and estimation because of the mismatching between received preamble and the reference preamble. When  $\Delta p = 0$ ,  $s_n^{(\Delta p)}$  is simply denoted as  $s_n$ .

In (3),  $\delta$  denotes the normalized frequency offset  $\delta = f_d T_s$ . This normalized offset will be estimated and is of relevance for the demodulator. However, for comparing estimation accuracy for different sampling rates we will normalize with respect to the symbol period  $T$ . For example, the simulation results for frequency estimate in Section V show the mean-squared error of  $\frac{T_s}{T} \delta = M\delta$ . Besides that,  $E_s/N_0$  represents the ratio of signal energy to noise power spectral density (SNR). Specifically,  $E_s$  is the averaged symbol energy of the received signal over the length of the preamble.

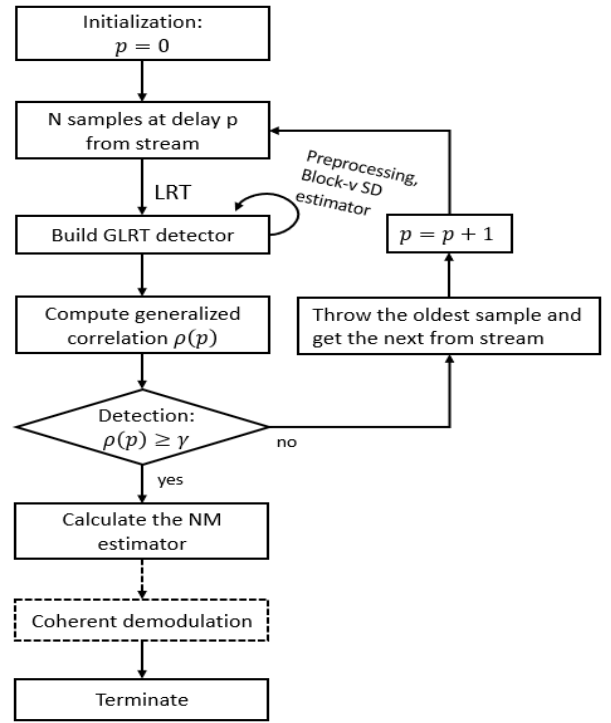


Fig. 1. The block diagram for complete signal acquisition chain in this paper

The rest of paper is separated in two main sections. The first section (Section III and IV) focus on analyzing the complete signal acquisition chain, which includes delay estimation of the preamble (detection) and carrier synchronization. The process is shown in Figure 1. The simulation section (Section V) then illustrates the performance of the proposed algorithm in first section. The second section (Section VI) moves attention on implementing the algorithm on software-defined radio (SDR). Some steps (equations) of proposed algorithm in first section are computed more efficiently to achieve the best throughput.

### III. DETECTION AND TIME SYNCHRONIZATION

As illustrated in Figure 1, the first step to analyze the signal acquisition chain is to formulate the detector. The detection algorithm proceeds sequentially and each step a window of length  $N$  received samples with  $N-1$  overlapped samples from the previous window is considered. The sequential detection problem solved in this paper is fundamentally equivalent to sequential frame synchronization [15]–[17]. To make analysis easier, we focus on deriving the detection algorithm by assuming the fractional delay  $\Delta p$  in (3) is neglected at a sufficiently high sample rate. The effect of  $\Delta p$  on our propose detection algorithm will be simulated with different sampling rate, or equivalently, different oversampling factor  $M$  in Section V.

We start by looking at the likelihood ratio test (LRT) for the detection task: Let  $H_0$  be the null hypothesis that the preamble is not completely presented in the received sequence from the observation window against the alternative  $H_1$  that it does. Define  $\Delta$  to be the distance between current received sequence

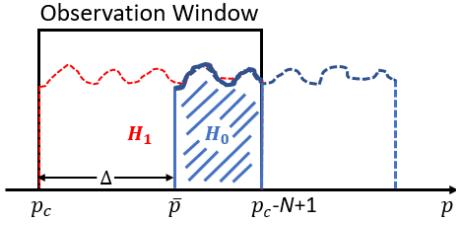


Fig. 2. Received sequence in observation window (the shaded area) containing partial preamble (Blue: current received sequence; Red: received sequence at true delay  $\bar{p}$ . The noise level is omitted)

at position  $p_c$  and the true position ( $\bar{p}$ ) of the preamble, i.e.,  $\Delta = |\bar{p} - p_c|$ . Figure 2 demonstrates the situation when received sequence containing the preamble has not completely passed the observation window. It is symmetric if the partial preamble shows on the left side, i.e., has passed the observation window ( $p_c > \bar{p}$ ). It is obvious that under  $H_0$ ,  $\Delta \neq 0$  while  $H_1$  means  $\Delta = 0$ . Based on (3), the conditional likelihood ratio test (LRT) can be built between  $H_0$  and  $H_1$  by giving the distance  $\Delta$ , the phasor  $S = Ae^{j\phi}$  and frequency offset  $b = e^{j2\pi\delta}$  at true delay  $\bar{p}$ ,

$$\begin{aligned} \Lambda(R|S, b, \Delta) &= \frac{p_{R|H_1, S, b}(r|H_1, S, b)}{p_{R|H_0, S, b, \Delta}(r|H_0, S, b, \Delta)} \\ &= \frac{\prod_{n=0}^{N-1} \frac{1}{\sqrt{\pi N_0}} \exp\left\{-\frac{1}{2} \frac{|r_n - s_n S b^n|^2}{N_0/2}\right\}}{\prod_{n=\Delta}^{N-1} \frac{1}{\sqrt{\pi N_0}} \exp\left\{-\frac{1}{2} \frac{|r_n - s_{n-\Delta} S b^{n-\Delta}|^2}{N_0/2}\right\}} \times \\ &\quad \frac{1}{\prod_{n=0}^{\Delta-1} \frac{1}{\sqrt{\pi N_0}} \exp\left\{-\frac{1}{2} \frac{|r_n|^2}{N_0/2}\right\}} \stackrel{H_1}{\gtrless} \eta. \end{aligned} \quad (5)$$

Cancelling the common parts and taking the logarithm, (5) is reduced to

$$\begin{aligned} \Re \left\{ \sum_{n=0}^{N-1} r_n s_n^* S^* b^{-n} - \sum_{n=\Delta}^{N-1} r_n s_{n-\Delta}^* S^* b^{-(n-\Delta)} \right\} &\stackrel{H_1}{\gtrless} \frac{N_0}{2} \ln \eta \\ &+ \frac{A^2}{2} \sum_{n=N-\Delta}^{N-1} |s_n|^2. \end{aligned} \quad (6)$$

The second summation on the left hand side of (6) points to the inner product of the overlap between the observed (partial) preamble and the true preamble. Or, to put it another way, (6) says that the log-likelihood ratio test equals to the difference between full cross-correlation and overlapped cross-correlation between received and reference sequences. However, note that in practice, the latter cannot be precomputed during detection process since  $\Delta$  is an unknown information to the receiver. One way to solving this problem is to simply move the overlapped cross-correlation to the right hand side and let it be reflected by the threshold. Thus, based on the above discussion, after

some proper scaling, the LRT finally reduces to a generalized correlation function in terms of the time instant (delay)  $p$ ,

$$\rho(p) = \frac{\Re\{\langle \mathbf{r}_p, \hat{\mathbf{s}}_p \rangle\}}{\|\mathbf{r}_p\| \cdot \|\hat{\mathbf{s}}_p\|} \stackrel{H_1}{\gtrless} \gamma \quad (7)$$

where  $\hat{\mathbf{s}}_p$  denotes the carrier-estimate corrected preamble, i.e.,  $\hat{s}_p[n] = s_n \hat{S}_p \hat{b}_p^n$  for  $n = 0, \dots, N-1$ , at delay  $p_c$ .  $\hat{b}_{p_c}$  and  $\hat{S}_{p_c}$  are the frequency and phasor estimates at delay  $p_c$ .  $\|\mathbf{r}_p\|$  is the Euclidean norm of the received signal at delay  $p$ .  $\gamma$  is the normalized detection threshold which lies on the range of  $[0, 1]$ . Note, the LRT in (5) is not practical since the frequency and phasor offsets at true delay  $\bar{p}$  is unknown while detection as well as the distance  $\Delta$ . Thus, instead of LRT, a generalized likelihood ratio test (GLRT) based detector is built relying on frequency and phasor estimates from each window of received sequence; To get  $\hat{b}_{p_c}$  and  $\hat{S}_{p_c}$ , recall the LRT of (5) is built conditionally on the known frequency and phase offset at the true position  $\bar{p}$  of the preamble; Therefore, the algorithm of computing the frequency and phase estimates at  $p_c$  can be also derived by assuming the current received sequence contains the complete preamble.

In conclusion, a GLRT based detector in (7) is derived for detection task in this paper, and it relies on the frequency and phase estimates of the observed data sequence. Furthermore, since the sequential detection proceeds at every time instant, to make it work at high sampling rate, the complexity of designed estimator for building (7) becomes much crucial. An estimator with very low computational complexity should be derived for real detection purpose.

#### IV. FREQUENCY AND PHASE ESTIMATION

Frequency and phase estimation, also called carrier synchronization, is the next step after detection for coherent demodulation. In this section, we discuss the estimation algorithm by assuming the time synchronization of the preamble is perfect, i.e., the observation window contains the complete preamble. For the same reason as in the detection section, we first derive the estimator by assuming the effect of fractional delay  $\Delta p$  is neglected. Later in simulation section, we will show how much the value of fractional delay degrades the estimating accuracy of estimators by comparing with different sampling rate.

For estimating frequency offset  $\delta$  and the phasor  $S = Ae^{j\phi}$ , the maximum likelihood (ML) estimation of the parameters in (3) is given by

$$\hat{\delta}, \hat{S} = \min_{\delta, S=Ae^{j\phi}} \sum_{n=0}^{N-1} |r_n - s_n S e^{j2\pi\delta n}|^2. \quad (8)$$

By taking the Wirtinger derivative with respect to  $S$  and setting it equal to zero, a closed form for the estimated phasor  $\hat{S}$  is readily derived,

$$\hat{S} = \frac{\sum_{n=0}^{N-1} r_n s_n^* e^{-j2\pi\delta n}}{\sum_{n=0}^{N-1} |s_n|^2}, \quad (9)$$

and  $\hat{\phi} = \arg\{\hat{S}\}$ . It can be seen the estimate of phasor  $\hat{S}$  relies on the estimate of frequency  $\hat{\delta}$ . Thus, the frequency and phasor

offset estimates are joint estimates. Moreover, by plugging (9) into (7), changing from taking the real value by absolute value, an approximate GLRT based detector finally reduces to

$$\rho(p) \approx \frac{|\tilde{S}_p|}{\|\mathbf{r}_p\| \cdot \|\mathbf{s}\|} \frac{H_1}{H_0} \gtrless \gamma \quad (10)$$

where  $\tilde{S}$  denotes the frequency-corrected cross-correlation (the numerator) of phasor estimate in (9), and  $\|\mathbf{s}\|$  is the Euclidean norm of the preamble. Note, compared with (7), (10) greatly lowers the computational complexity of the detector.

The frequency estimate is obtained similarly as the zero of the derivative of (8),

$$\sum_{n=0}^{N-1} (r_n s_n^* S^* n e^{-j2\pi\delta n} - s_n s_n^* n) = 0. \quad (11)$$

Note  $\sum_{n=0}^{N-1} s_n s_n^* n$  is real valued, which results in the imaginary part of left hand side of (11) be zero; By plugging the estimate  $\hat{S}$  of (9) into (11) and rearranging the order of indexes, yields

$$\Im \left\{ \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} n r_n r_m^* s_m^* s_n e^{j2\pi\delta(m-n)} \right\} = 0. \quad (12)$$

A change of variables let us focus on the difference between sampling instances  $m$  and  $n$ . Defining  $k=m-n$ , (12) becomes

$$\Im \left\{ \sum_{m=0}^{N-1} \sum_{k=m-(N-1)}^m (m-k) r_{m-k} r_m^* s_{m-k}^* s_m e^{j2\pi\delta k} \right\} = 0. \quad (13)$$

Reversing the order of summation in (13), we get

$$\Im \left\{ \sum_{k=-(N-1)}^0 \sum_{m=0}^{N-1+k} (m-k) r_{m-k} r_m^* s_{m-k}^* s_m e^{j2\pi\delta k} + \sum_{k=1}^{N-1} \sum_{m=k}^{N-1} (m-k) r_{m-k} r_m^* s_{m-k}^* s_m e^{j2\pi\delta k} \right\} = 0. \quad (14)$$

Note, the term for  $k=0$  in (14) can be eliminated since it is real-valued. For  $k \neq 0$ , the positive and negative indices  $k$  are symmetric. After grouping terms appropriately, the necessary condition for  $\hat{\delta}$  is given by

$$J(\hat{\delta}) = \Im \left\{ \sum_{k=1}^{N-1} \sum_{m=k}^{N-1} k r_{m-k} r_m^* s_{m-k}^* s_m e^{j2\pi\hat{\delta}k} \right\} = 0. \quad (15)$$

This expression is fundamentally equivalent to conditions provided by Luise and Reggiannini [6] and Fitz [7]. However, (15) explicitly allows for pulse shaping and oversampling. Note, the estimator  $\hat{\delta}$  in (15) has no closed-form solution. In [6], it is approximated by replacing the exponential with its Taylor series expansion. In [7], an approximate solution is obtained via Euler's identity for large  $N$ . Both solutions have computational complexity  $O(N^2)$  reflecting the double summation.

Since our paper focus on the joint detection and estimation problem, we propose a family of alternative solutions to (15).

A coarse solution with  $O(N)$  complexity is used for operating at high sampling rate during the sequential GLRT detection; It prioritizes the low complexity at the expense of some loss of accuracy. A fine solution is used to improve the estimation accuracy for the coherent demodulation once the preamble has been detected. The two solutions are illustrated with details in the next sections.

#### A. Solution 1: (Coarse) Single-Difference (SD) Estimator

The first estimator is rooted in the insight that at high SNR environment, every lag  $k$  in (15) can be used to approximate the true frequency offset  $\bar{\delta}$ . Assume noise is small, i.e.,  $r_m \approx s_m A e^{j(2\pi\delta m + \phi)}$ , (15) can be expanded to

$$\Im \left\{ A^2 \sum_{k=1}^{N-1} \sum_{m=k}^{N-1} k |s_{m-k}|^2 |s_m|^2 e^{j2\pi(\hat{\delta}-\bar{\delta})k} \right\} = 0. \quad (16)$$

Note that in (16) the inner summation is purely real for every lag  $k$  if  $\hat{\delta} = \bar{\delta}$ . This suggests that an unbiased estimate of frequency offset  $\hat{\delta}$  can be obtained by using only a single lag  $k$  from (15). The approach lowers the complexity from  $O(N^2)$  to  $O(N)$  and permits a closed-form solution for  $\hat{\delta}$ . While, the disadvantage of the estimator by using single lag  $k$  is its insufficient estimating accuracy since it lacks of the processing gain by outer integrator. Thus, we name the estimator as the coarse Single-Difference (SD) estimator. We just omit "coarse" for simplicity from now. Just because of the low complexity and moderate estimating accuracy, the SD estimator is used for frequency correction of sequential GLRT detector in (7).

1) *Closed-form expression:* For one lag  $k$ , we first give the expression of the (primary) SD estimator that can be directly obtained by reducing (16),

$$\hat{\delta}_{\text{primary-SD}}(k) = -\frac{\arg \left\{ \sum_{m=k}^{N-1} r_{m-k} r_m^* s_{m-k}^* s_m \right\}}{2\pi k}. \quad (17)$$

2) *Performance of the primary SD estimator:* In low SNR environment, i.e., noise effect cannot be ignored, the argument of numerator, denoted as  $W(k)$ , in (17) can be extended to

$$\begin{aligned} W(k) &= \sum_{m=k}^{N-1} r_{m-k} r_m^* s_{m-k}^* s_m \\ &= \sum_{m=k}^{N-1} \left( A^2 |s_{m-k}|^2 |s_m|^2 e^{-j2\pi\bar{\delta}k} + w_m^* S |s_{m-k}|^2 s_m e^{j2\pi\bar{\delta}(m-k)} + \right. \\ &\quad \left. w_{m-k} S^* |s_m|^2 s_{m-k}^* e^{-j2\pi\bar{\delta}m} + w_{m-k} w_m^* s_{m-k}^* s_m \right). \end{aligned} \quad (18)$$

To interpret (18), recognize that the first term of right hand side is deterministic and provides the mean of the expression. The middle two terms yield a zero-mean, complex Gaussian random variable; While, the last term performs another zero-mean random variable with a second kind Bessel distribution. Note, the two random variables are uncorrelated. Moreover, we assume  $N-k$  is large, by central limit theorem, the second kind Bessel random variable is considered approximately Gaussian. Thus, (18) is summarized as

$$W(k) \sim \mathcal{CN}\left(\left(\frac{N-k}{A^2}\right)\left(\frac{E_s}{M}\right)^2 e^{-j2\pi\delta k}, 2\left(\frac{N-k}{A^4}\right)\frac{N_0}{2}\left(\frac{E_s}{M}\right)^3 + \left(\frac{N-k}{A^4}\right)\left(\frac{N_0}{2}\right)^2\left(\frac{E_s}{M}\right)^2\right), \quad (19)$$

where  $A^2|s_m|^2 \approx E_s/M$  denotes the average energy per sample. A quick test for the primary SD estimator with respect to SNR is to look at the ratio of (absolute) square of mean to variance of  $W(k)$  in (19), which is denoted as the output SNR,

$$\begin{aligned} \text{SNR}_{\text{out\_primary}} &= \frac{|\mu_{W(k)}|^2}{\sigma_{W(k)}^2} = \frac{N-k}{2 \cdot \frac{N_0}{2} \left/ \frac{E_s}{M} + \left(\frac{N_0}{2}\right)^2 \left/ \left(\frac{E_s}{M}\right)^2 \right.} \\ &= \frac{N-k}{2/\text{SNR}_{\text{in}} + 1/\text{SNR}_{\text{in}}^2} \end{aligned} \quad (20)$$

where  $\mu_{W(k)}$  and  $\sigma_{W(k)}^2$  denote the mean and variance of  $W(k)$  provided in (19), and  $\text{SNR}_{\text{in}} = \frac{E_s}{M} / \frac{N_0}{2}$  is defined as the average input sample energy to noise power spectral density. In (20),  $N-k$ , which is the size of integrator in (18), exactly provides the processing gain; At relatively high SNR, the inverse-square input SNR, which is from the variance of second kind Bessel variable, could be neglected; Thus, the estimating accuracy of primary SD estimator exhibits a linear relationship with high  $\text{SNR}_{\text{in}}$ . At low input SNR, the inverse-square SNR dominates and degrades the  $\text{SNR}_{\text{out}}$  of  $W(k)$  quadratically, which makes the  $\text{SNR}_{\text{out}}$  fast approach and then be lower than the minimum requirement that maintains the required accuracy for building detector and later a fine estimator.

Now, we will derive the distribution of primary SD estimator. Recall from (17), the distribution of  $\hat{\delta}_{\text{primary-SD}}(k)$  requires  $\arg\{W(k)\}$ . The complete probability density function (pdf) of  $\arg\{W(k)\}$  is derived in appendix A, where it also shows that a good approximation, valid for moderate SNR, is Gaussian. Specifically

$$\arg\{W(k)\} \sim \mathcal{N}\left(\angle\mu_{W(k)}, \frac{\sigma_{W(k)}^2}{|\mu_{W(k)}|^2}\right). \quad (21)$$

Thus, (21) proves the variance of primary SD estimator is inverse proportional to  $\text{SNR}_{\text{out}}$  in (20). By plugging (19) in (21) without adding second Bessel variance term, the primary SD estimator of (17) is approximately Gaussian distributed at moderate SNR with pdf

$$\hat{\delta}_{\text{primary-SD}}(k) \sim \mathcal{N}\left(\bar{\delta}, \frac{M}{4\pi^2 k^2 (N-k) E_s / N_0}\right). \quad (22)$$

We see that  $\hat{\delta}_{\text{primary-SD}}(k)$  is unbiased. Note, the distribution of primary SD estimator in (22) only exploits a good fit for moderate SNR due to the constraint of (21). A rough calculation for distribution at low SNR is to simply replace the variance of Gaussian by variance of second kind Bessel of (19) and plugging in (21). The resulting variance, or equivalently, the mean-squared error (MSE) of the primary SD estimator at

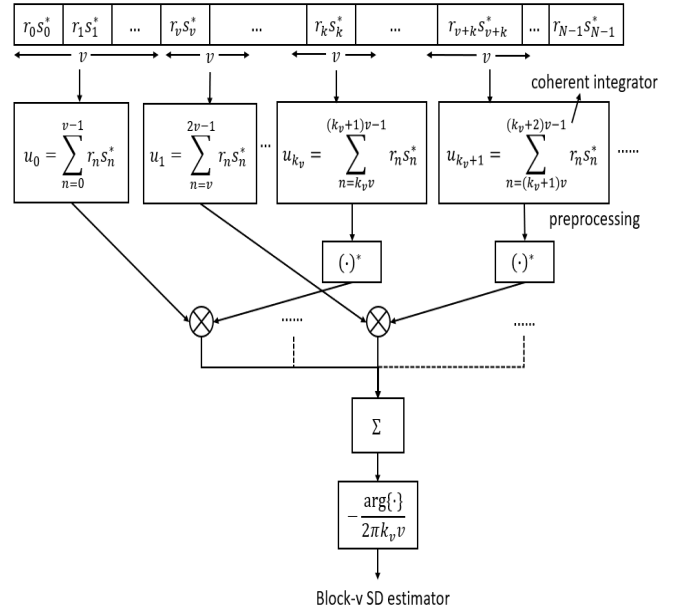


Fig. 3. Flow diagram of implementing the Block- $v$  SD estimator

low SNR, compared to high SNR, is increased by  $\frac{M}{4E_s/N_0}$ . The solution to mitigating the increased variance of the primary SD estimator at low SNR is given in the next section.

3) *Block- $v$  SD estimator*: The reason for the primary SD estimator of (17) has a relatively bad performance at low SNR is because the large noise disturbance significantly decorrelates the received signal and the preamble at each sample instant. An alternative SD estimator is proposed by changing the order of computation in (17). Specifically, instead of calculating the correlation at each sample instant, cross-correlation between  $k$  sample instant and then averaging, we first average the correlations for some coherent time instants; By doing this, some of processing gain is applied to increase the estimating accuracy in every small section of sample instants. The results are called coherent integrators and the process is called preprocessing. After that, use the remaining processing gain to integrate the coherent integrators between non-coherent time instants. The details are illustrated in Figure 3.

In Figure 3, the coherent integrator  $u_l$  is formulated by

$$u_l = \sum_{n=l_v}^{(l+1)v-1} r_n s_n^*, \quad \text{for } l = 0, 1, \dots, N/v-1 \quad (23)$$

where  $v$  denotes the number of coherent correlations that are for averaging. The value of  $v$  is set to be a factor of  $N$  to include all sample instants.  $k_v$  denotes the distance between coherent integrators that calculates correlation of non-coherent time instants. Thus,  $k_v = \lfloor k/v \rfloor$ , where  $\lfloor \cdot \rfloor$  represents the floor operation, since  $v$  is not necessary to be a factor of  $k$ . However, we still assume  $k \approx k_v v$  for comparison simplicity. Based on above discussion, the alternative Block- $v$  SD estimator is given by

$$\hat{\delta}_{\text{Block-}v\text{-SD}}(v, k_v) = -\frac{\arg \left\{ \sum_{l=k_v}^{N/v-1} u_l^* u_{l-k_v} \right\}}{2\pi k_v v} \quad (24)$$

It can be seen that the primary SD estimator in (17) is a special case of Block- $v$  SD estimator when  $v = 1$ . Thus, we uniformly call the two estimators as the Block- $v$  SD estimator.

Now, we discuss the performance of Block- $v$  SD estimator by focusing on the cross-correlation between the two general coherent integrators in (24). Based on (23) and the derivation of (19), the coherent integrator  $u_l^*$  is complex Gaussian distributed with probability density function,

$$u_l^* \sim \mathcal{CN} \left( \frac{E_s/M S^*}{A^2} \sum_{n=l_v}^{(l+1)v-1} e^{-j2\pi\delta n}, v \frac{E_s/M N_0}{A^2} \frac{N_0}{2} \right),$$

and the pdf of complex Gaussian-distributed  $u_{l-k_v}$  is also given by

$$u_{l-k_v} \sim \mathcal{CN} \left( \frac{E_s/M S}{A^2} \sum_{n=(l-k_v)v}^{(l-k_v+1)v-1} e^{j2\pi\delta n}, v \frac{E_s/M N_0}{A^2} \frac{N_0}{2} \right).$$

Note,  $u_l^*$  and  $u_{l-k}$  are uncorrelated. Compared with (18), the cross-correlation between two coherent integrates  $u_l^*$  and  $u_{l-k_v}$  also yields a random variable with a combination of Gaussian and second kind Bessel function. After some algebra, the mean and variance of  $u_l^* u_{l-k_v}$ , denoted as  $\mu_{C_u}$  and  $\sigma_{C_u}^2$ , can be simplified by, respectively,

$$\begin{aligned} \mu_{C_u} &= \frac{(E_s/M)^2}{A^2} \sum_{n=l_v}^{(l+1)v-1} e^{-j2\pi\delta n} \left( \sum_{m=(l-k)v}^{(l-k+1)v-1} e^{j2\pi\delta m} \right) \\ &= \frac{(E_s/M)^2}{A^2} e^{-j2\pi\delta k_v v} \left( \frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \right)^2, \\ \sigma_{C_u}^2 &= v^2 \frac{(E_s/M)^2}{A^4} \left( \frac{N_0}{2} \right)^2 + 2v \frac{(E_s/M)^3}{A^4} \frac{N_0}{2} \left( \frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \right)^2, \end{aligned} \quad (25)$$

where  $\frac{\sin(\pi\delta v)}{\sin(\pi\delta)}$  is one of the Dirichlet function, which approaches the maximum value  $v$  at  $\delta = 0$  and the first pair of zero crossings at  $\delta = \pm 1/v$ . Based on (20) and (25), the output SNR of Block- $v$  SD estimator in terms of the input SNR yields

$$\begin{aligned} \text{SNR}_{\text{out\_Block-}v} &= \frac{|\mu_{C_u}|^2}{\sigma_{C_u}^2} \\ &= \frac{\frac{(E_s/M)^4}{A^4} \left( \frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \right)^4 (N/v - k_v)^2}{\left[ v^2 \frac{(E_s/M)^2}{A^4} \left( \frac{N_0}{2} \right)^2 + 2v \frac{(E_s/M)^3}{A^4} \frac{N_0}{2} \left( \frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \right)^2 \right] (N/v - k_v)} \\ &= \frac{(N/v - k_v) \left( \frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \right)^4}{\frac{v^2}{\text{SNR}_{\text{in}}^2} + \frac{2v}{\text{SNR}_{\text{in}}} \left( \frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \right)^2}. \end{aligned} \quad (26)$$

Some important observations of  $\text{SNR}_{\text{out\_Block-}v}$  from (26) can be discussed. First, the  $\text{SNR}_{\text{out}}$  depends on the value of  $|\bar{\delta}|v$ . We begin to consider the case when  $|\bar{\delta}|v \ll 1$ , i.e.,  $\frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \approx v$  and  $(N/v - k_v)v \approx N - k$ . When the input SNR is moderate, the second term of variance dominates and results the nearly same output SNR as in (20); While at low SNR, the first term of variance dominates and ideally produces an approximate  $v$  times larger output SNR than the primary SD estimator, which corresponds to the design intention of block- $v$  ( $v \geq 2$ ) SD estimator. The selection of  $v$  trades off the processing gain at low SNR and incurred large frequency offset.

In practice,  $|\bar{\delta}|$  is generally not large, a reasonable assumption is to assume  $\bar{\delta}$  is in the range of  $[-1/v, 1/v]$ . By comparing (20) and (26), it is easy to get the Block- $v$  SD estimator produces a positive processing gain at low SNR when  $|\bar{\delta}|v$  satisfies  $\frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \in [v^{3/4}, v]$ ; However, The worst case happens when  $|\bar{\delta}|v_{\min}$  even also produces a relative large frequency offset, i.e.,  $v_{\min} = 2$  and  $\frac{\sin(2\pi\delta)}{\sin(\pi\delta)} \in [0, 2^{3/4}]$ ; The Block- $v$  SD estimator will not improve the performance than the primary SD estimator at low SNRs.

Now, let's measure the distribution of Block- $v$  SD estimator at low SNR. Assume  $|\bar{\delta}|v$  always satisfies  $v^{3/4} \leq \frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \leq v$ , and the extra processing gain  $v$  at relatively low SNR increases the  $\text{SNR}_{\text{out\_Blv}}$  to meet the moderate SNR requirement of (21). Thus, by plugging (25), (21) into (24), the block- $v$  SD estimator is distributed approximately Gaussian at relatively low SNR with probability density function

$$\begin{aligned} \hat{\delta}_{\text{Block-}v\text{-SD}}(v, k_v) &\sim \mathcal{N} \left( \bar{\delta}, \frac{M^2 v^2}{16\pi^2 k^2 \left( \frac{\sin(\pi\delta v)}{\sin(\pi\delta)} \right)^4 (N/v - k_v) (E_s/N_0)^2} \right). \end{aligned} \quad (27)$$

Thus,  $\hat{\delta}_{\text{Block-}v\text{-SD}}$  is also unbiased. Moreover, the lag  $k_v$  also affects the variance of Block- $v$  SD estimator. The best choice is to choose  $k_v = \lfloor \frac{2N}{3v} \rfloor$  to minimize the variance. However, it should be noted that if  $k_v v$  is too large, the frequency estimate  $\hat{\delta}_{\text{Block-}v\text{-SD}}$  may suffer the effect of "aliasing" if  $2\pi|\bar{\delta}|k_v v > \pi$ . The ambiguity problems are common in frequency estimation. The effective method to solve it is based on the robust Chinese reminder theorem.

It should be also compared and emphasized the complexity of block- $v$  ( $v \geq 2$ ) SD estimator with block-1 SD estimator in sequential detection process. The two estimators have the same complexity for single detection but large computational-complexity difference in sequential detection. Note, for block- $v$  ( $v \geq 2$ ) SD estimator, because of preprocessing, each sample in received sequence should be multiplied by the sample in the preamble at same sample instant; This greatly limits the way of computing between two adjacent sequential detection. However, for block-1 SD estimator in (17), the more efficient way is to calculate the autocorrelations for both received and reference sequence with sample difference  $k$  and then multiply. By doing this method, except for the first received sequence, we only need to calculate one more autocorrelation for the

incoming sample at each detection. Since the autocorrelations of samples in the preamble can be pre-computed, the argument of (17) can be computed by the cross-correlation between the autocorrelations of received sequence in a shift register and autocorrelations of reference samples in a fixed vector. Thus, compared with block- $v$ , block-1 SD estimator greatly reduces the computational redundancy in sequential detection.

### B. Solution II: (Fine) Newton-Method (NM) Estimator

The (block- $v$ ) SD estimator emphasizes the low-complexity property and is intended to provide merely sufficiently good carrier synchronization to enable coherent detection. Once the signal has been acquired, the accuracy of the block- $v$  SD estimator can be improved by investing additional computations. Since detection events are rare, the computational complexity is of little concern.

The principle is to use the Block- $v$  SD estimator as the starting point for a Newton-type iteration aimed at finding a better solution to the necessary condition (15). In principle, multiple iterations are possible to produce successively better approximations to the root of  $J(\hat{\delta})$  in (15). Specifically, the iterations are given by

$$\hat{\delta}_{\text{NM}}^{(i+1)} = \hat{\delta}_{\text{NM}}^{(i)} - \frac{J(\hat{\delta}_{\text{NM}}^{(i)})}{J'(\hat{\delta}_{\text{NM}}^{(i)})} \quad (28)$$

where  $\hat{\delta}_{\text{NM}}^{(0)} = \hat{\delta}_{\text{SD}}(k_{\text{opt}})$  is the starting point of the iteration and  $J'(\cdot)$  denotes the derivative of  $J$  with respect to  $\hat{\delta}$ . Specifically,

$$J'(\hat{\delta}) = \Im \left\{ \sum_{k=1}^{N-1} \sum_{m=k}^{N-1} j2\pi k^2 r_{m-k}^* s_{m-k}^* s_m e^{j2\pi \hat{\delta} k} \right\}. \quad (29)$$

Our simulations indicate that only a single iteration is usually sufficient to achieve very good accuracy.

## V. SIMULATION RESULTS

The simulation section illustrates the results in the order of detection, estimation and joint detection and estimation. Four different lengths of reference symbol sequences are simulated with 50% rolloff Square-Root Raised Cosine (SRRC) pulses. The reference sample sequence is chosen from Gold sequence and modulated by a QPSK alphabet with good autocorrelation properties. We assume the normalized frequency offset is small enough so that the design parameter  $k$  of the SD estimator can be chosen optimally by  $2/3$  length of the preamble (For  $K$ -SD estimator, all the estimates of SD satisfying the aliasing limits).

### A. Simulation Results for Detection

In this section, we first illustrate the simulation results of the detection algorithm without considering the fractional delay. The effect of fractional delay on detection will be discussed at the end of this section.

Figure 4 illustrates the performance of the proposed detector in (7) without fractional delay for  $L_0 = 32$ ,  $M = 8$ , 0 dB SNR. The GLRT based detector performs robust at low SNR since it achieves the highest generalized correlation at  $\bar{p}$ . However,

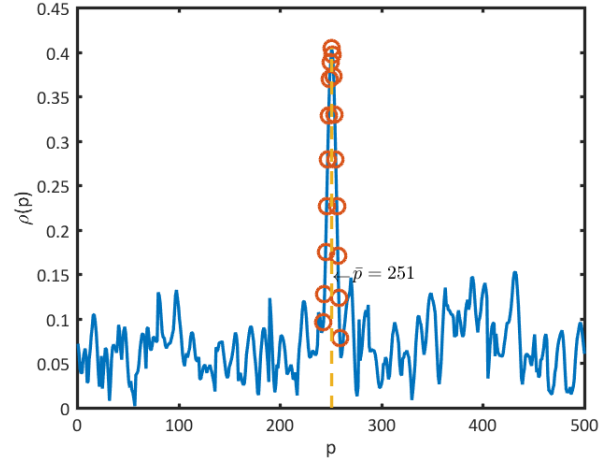


Fig. 4. Performance of GLRT detector of (7) at each delay of received stream (SNR= 0 dB, no fractional delay, dashed line: the position of  $\bar{p}$ )

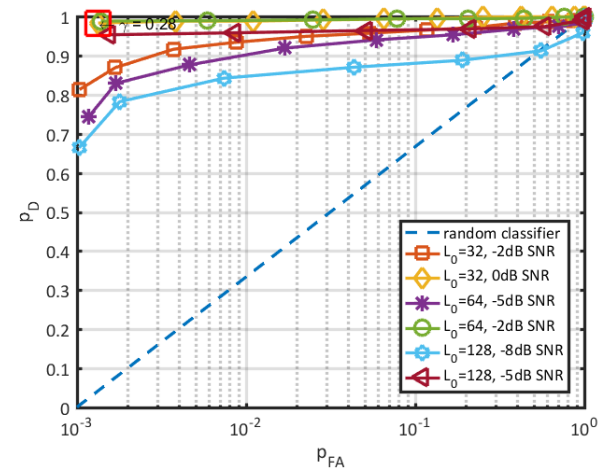


Fig. 5. Receiver operating characteristics of GLRT based detector for different lengths of reference sequence and SNR

from the adjacent correlations centered at  $\bar{p}$  that we point out, the correlation decays very slow and thus it will make much more challenging to choose the threshold to distinguish between the true position of received sequence and its corresponding adjacent positions. This is due to pulse shaping and oversampling. Specifically, the width of "mainlobe" depends on the shape of pulse and the value of  $M$ , e.g., it will become wider if  $M$  is larger. To accommodate this imperfect autocorrelation influence by pulse shaping and oversampling, we adjust the detection algorithm by finding the local maximum of the correlation near  $\bar{p}$  instead of just comparing the correlations with threshold to make the decisions at each delay.

Figure 5 shows the receiver operating characteristics (ROC) of the proposed (adjusted) detector. It basically illustrates that the detector has good ROCs at very low SNR. Specifically, the detector achieves perfect ROCs for positive SNR for all length of  $L_0$ ; Moreover, it is still robust to negative SNR environment.



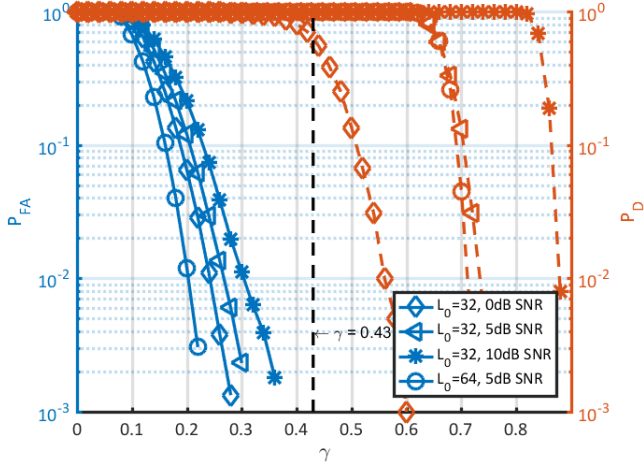


Fig. 6. False alarm and detection ratio for different SNR and sizes of preamble ( $M = 4$ . Blue: false alarm probability. Red: detection probability)

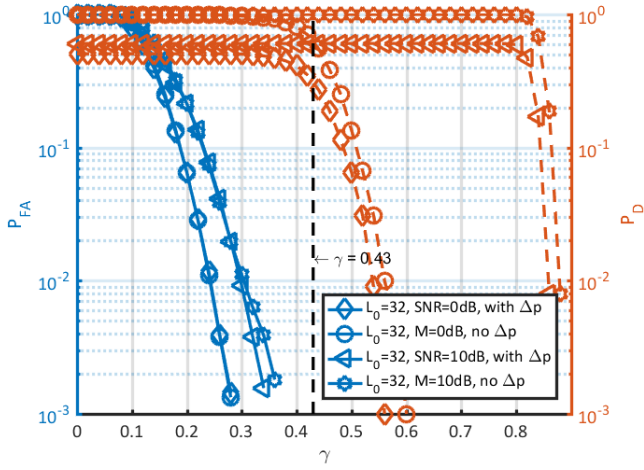


Fig. 7. Comparison between false alarm and detection ratio with and without maximum  $\Delta p$  for same  $L_0$  and different SNR ( $M = 4$ . Blue:  $P_{FA}$ . Red:  $P_D$ )

For instants, we can select such a threshold to achieve  $P_{FA} \approx 10^{-3}$  and  $P_D \geq 0.8$  simultaneously when  $L_0 = 32$  and SNR = -2 dB. However, it is not clear from this figure to pick an exact threshold for certain detection purpose.

Figure 6 shows the performance of detector but from another perspective. Specifically, it gives us a deep insight into the false alarm and detection probabilities versus the threshold. Basically, we have two observations. First, compared with the two curves of 0 dB and 10 dB SNR for  $L_0 = 32$ , we see the latter needs to achieve the same false alarm probability with a higher threshold. This is because the level of "sidelobe" in Figure 4 is increasing in terms of SNR; If we look at the same two curves of  $P_D$ , the curve of a high SNR maintains a perfect  $P_D$  at a higher threshold. This is due to the increasing level of "mainlobe" via SNR. Second, compared with two curves with different  $L_0$  and same SNR, we see that the received sequence with more symbols achieves the same  $P_{FA}$  at lower threshold.

The reason is that the sequence with more symbols will have a better autocorrelation property. Moreover, it is commonly hard to pick such a threshold to accommodate all specific detection purposes. For simulation purpose, now if we want to determine a threshold to work for all reference sequences with  $L_0 \geq 32$  and SNR up to 10 dB, based on Figure 6 and Neyman-Pearson criterion,  $\gamma = 0.43$  is a good choice to meet  $P_{FA} < 1e^{-3}$  and nearly perfect  $P_D$  for all lengths of  $L_0$ .

To this point, we discuss the effect of fractional delay  $\Delta p$  in model (3) on modified detection algorithm. Above, we assume  $\Delta p$  is neglected when sampling rate  $f_s$  is sufficiently large. However, when  $f_s$  is not much bigger than symbol rate, or equivalently, the oversampling factor  $M$  is relatively small, the value of  $\Delta p$  will degrade the performance of detection and estimation due to mismatching between the preamble in received and reference sequence. For example, when  $M=1$  and the preamble has the maximum fractional delay equaling to  $\Delta p = \pm \frac{1}{2f_s} = \pm \frac{T}{2M} = \pm \frac{T}{2}$ , half of the preamble in received and reference sequence is mismatched thus the peak of generalized correlation in Figure 4 will disappear. Instead, Two peaks will show up: one at  $\bar{p}$  and another one at the adjacent  $p$  depending on the sign of  $\Delta p$ . Recall, the detection algorithm is to find the local maximum of generalized correlation around  $\bar{p}$ . Thus, there is no doubt that  $\Delta p$  will greatly decrease the detection probability, as shown in Figure 7.

In Figure 7, the most intuitive observation is that the curves of  $P_D$  with (maximum) fractional delay for all SNR do not start at 1 when  $\gamma$  is very low. The detector is hard to make the correct decision between the adjacent two delays with similar high correlations. If we compare with the two curves of  $P_D$  for SNR = 0 dB and 10 dB both with fractional delay, the initial  $P_D$  of 10 dB SNR is approximately 0.6 and greater than 0.5 at 0 dB. This is because at low SNR, the noise level degrades the detection performance; While, at high SNR, the percentage of the mismatched preamble between received and reference sequence is below 50% since  $\Delta p = \pm \frac{T}{2M} = \pm \frac{T}{8}$ . The generalized correlation at  $\bar{p}$  will be larger than at the adjacent delay with the probability more than 1/2.

We can also see the fractional delay has a small effect on false alarm probability, which is important. For example, as we discussed above, if we want to choose a threshold to work for the detection task that all reference sequences with  $L_0 \geq 32$  and SNR  $\leq 10$  dB, then from Figure 6 and 7,  $\gamma = 0.43$  is still a good choice to meet the requirement of false alarm probability. Note, when  $\Delta p$  is large, although  $\Delta p$  will "lead" the detector to make the decision between the two adjacent delays with similar highest generalized correlation, we can just keep either one as the detection result but care more about the estimation accuracy with the determined received sequence for coherent demodulation. Thus, in the next section, we will talk about the degradation of estimation accuracy with different levels of  $\Delta p$ . The goal is to choose the oversampling factor  $M$  that makes the estimation error negligible.



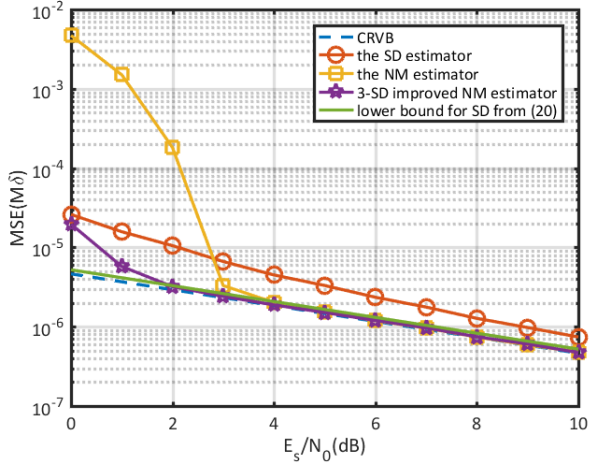


Fig. 8. Accuracy of the SD and SD (or  $K$ -SD) based NM estimators ( $L_0 = 32$ )

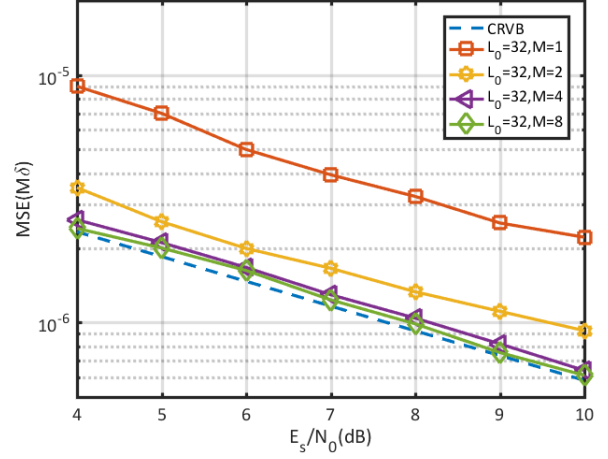


Fig. 10. Accuracy of NM estimator with maximum fraction delay for different value of oversampling factor ( $L_0 = 32$ ,  $\Delta p = \frac{T}{2M}$ )

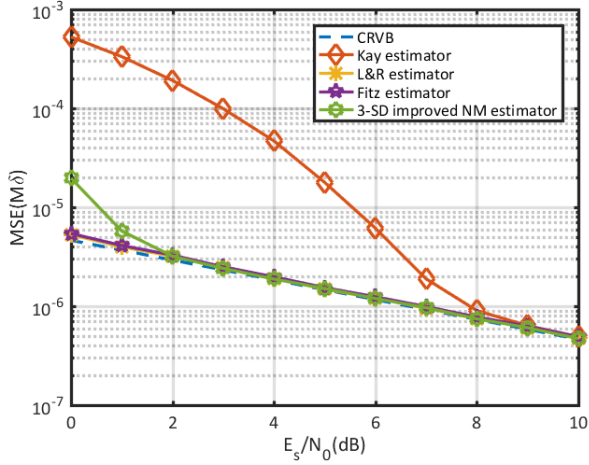


Fig. 9. Accuracy of the NM estimator and conventional estimators ( $L_0 = 32$ )

### B. Simulation Results for Estimation

In this section, we focus on the estimating accuracy of our proposed estimators (SD, or  $K$ -SD and NM) by assuming no fractional delay and the sequential GLRT detection progress is perfect. The simulation shows the NM estimator can achieve as good accuracy as those conventional estimators in [5]–[7] for coherent demodulation at moderate SNR. Then, the effect of fractional delay  $\Delta p$  on estimating accuracy with respect to  $M$  will be discussed. We find that  $M = 4$  yields a negligible estimation error from  $\Delta p$ . At the end of this section, we will also show the estimating accuracy of NM estimator after joint detection and simulation for coherent demodulation.

Figure 8 gives the insight into the accuracy of two proposed estimators. The lower dashed line denotes the Cramer-Rao vector bound (CRVB) for frequency estimate in (3) multiplied by  $M$ , which is given by

$$\text{CRVB}(M\delta) \geq \frac{3}{2\pi^2 L_0^3 E_s/N_0}. \quad (30)$$

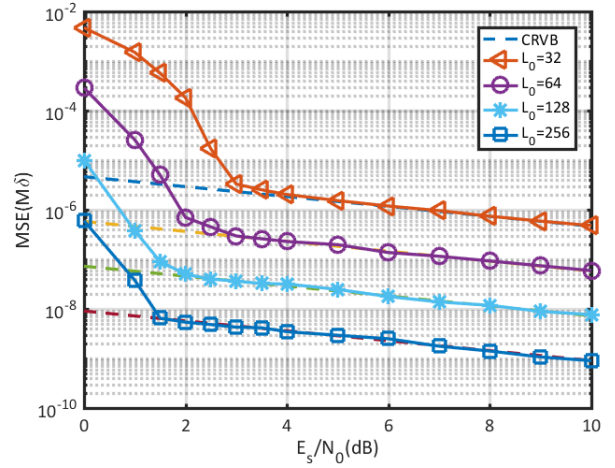


Fig. 11. Accuracy of NM frequency estimate in joint detection and estimation ( $\gamma = 0.43$ ,  $M = 4$ ,  $\Delta p \in (-\frac{T}{8}, \frac{T}{8}]$ )

Furthermore, the CRVB for  $\phi$  is derived as

$$\text{CRVB}(\phi) \geq \frac{2}{L_0 E_s/N_0}. \quad (31)$$

The derivation steps of (30) and (31) are given in Appendix B. In Figure 8, we see the NM estimator approaches the CRVB at SNR= 4 dB. Note, the accuracy of the SD estimator is crucial not only just for building the GLRT detector but deciding the accuracy of the NM estimator. The evidence is that the NM estimator performs even worse than the SD estimator at low SNR. This is because the Newton iteration of (28) converges occasionally to local minimum away from the true frequency offset if the initial (SD) estimate is far from the true frequency offset. Moreover, Figure 8 shows the averaging method of (??) improves the accuracy of the SD estimator thus improves the NM estimator at low SNR.

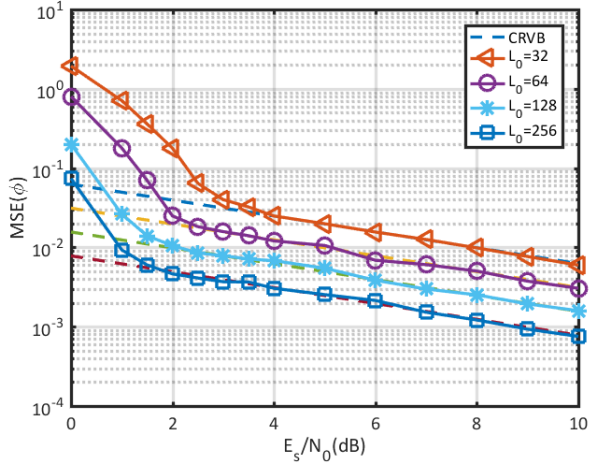


Fig. 12. Accuracy of the NM phase estimate in joint detection and estimation ( $\gamma = 0.43$ ,  $M = 4$ ,  $\Delta p \in (-\frac{T}{8}, \frac{T}{8})$ )

Figure 9 compares the performance of the NM estimator and conventional estimators in [5]–[7]. Related to Figure 8, we conclude that the NM estimator can achieve as good accuracy as those autocorrelation-based estimators (L&R estimator [6] and Fitz estimator [7]) when the SD estimator is accurate enough. Put it another way, without increasing complexity by averaging with multiple SD estimators, the NM estimator can achieve the same good accuracy as the traditional estimators at moderate SNR.

Now we are going to discuss the effect of  $\Delta p$  on estimating accuracy of the NM estimator. In the previous section, we got the conclusion that  $\Delta p$  both degrades the detection probability and estimating accuracy because of the mismatching between the preamble in received and reference sequence. The solution to dealing with the degradation of detection is to maintain the decision of detector but sacrifice the accuracy of the estimator. Thus, we need to select the oversampling factor  $M$  that makes the fractional delay  $\Delta p$  small enough and the estimation error negligible. Figure 10 illustrates the estimating accuracy of NM estimator with maximum fractional delay equaling to  $\Delta p = \frac{T}{2M}$  for different  $M$ . It can be seen that when  $M = 1, 2$ , i.e.,  $\Delta p = \frac{T}{2}, \frac{T}{4}$ , the gap between the mean-squared error (MSE) and the CRVB is obvious; The MSE of two curves for  $M = 4$  and  $M = 8$  both nearly approach the CRVB. But note, the reference sequence with  $M = 8$  induces an extra double computational complexity than  $M = 4$ . Thus,  $M = 4$  is an ideal choice for joint detection and estimation purpose in this paper.

At the end of simulation section, the complete signal acquisition chain (joint detection and estimation) is simulated. Figure 11 and 12 illustrate the performance of NM estimator after joint detection and estimation.  $\Delta p$  is included with the range determined by  $M$ . The parameters, e.g., the value of  $\gamma$ , the oversampling factor  $M$ , are chosen based on the previous discussion. No averaged SD estimator is used because we want to reduce the complexity of the sequential detection process. Compared with Figure 6 and Figure 8, the reason for the NM

estimator not approaching the CRVB at low SNR is due to non-sufficient accuracy of the SD estimator. Besides that, we can get a common conclusion that by increasing the size of the preamble, better performance of carrier synchronization is realized and CRVB is approached at lower SNR.

## VI. IMPLEMENTATION ON SOFTWARE-DEFINED RADIO

Two main aspects of measuring the performance of modern communication systems, are the latency and throughput. In the previous sections, we focus on explaining and showing how much accuracy of our proposed joint detection and estimation algorithm can be. Now in this section, we discuss how to realize our proposed algorithm on software-defined radio (SDR). The algorithm, especially the detection algorithm, should be refined to fitting very high sample rate since it is applied on every time instant.

### A. Modified algorithm in Threading Building Blocks (TBB)

At the receiver side, in order to make the algorithm work at a very high data rate, we need to increase the computation efficiency of our detection algorithm as much as possible. The first improvement is by using pipeline. Threading Building Blocks (TBB) is a well-known C++ library that enables parallel programming on multi-core processors [18], [19]. The Flow Graph interfaces in TBB [19, Ch. 3] can make the algorithm implemented in a simple structure and realize pipeline very well by separating the algorithm into small blocks (nodes), which is used in this paper.

Figure 13 illustrates the implementation of the proposed joint detection and estimation algorithm in flow graph of TBB. Each block corresponds to a function node in the flow graph. The received data stream is first buffered into small size of pieces. The size of buffer should be determined to balance the latency and throughput: an oversized buffer will cause a long latency while an undersized buffer incurs a large amount of overhead. Then, the data in buffer is processed for preparing the elements of building the GLRT detector, which denotes the dashed area. Now, we will go deep explaining the nodes in this area. The top level generates the SD estimators in a vector in terms of each discrete time instant. Specifically, the computation of  $\hat{\delta}_{SD}$  in (17) is separated into 3 steps (nodes): the first node includes a shift register and calculates the autocorrelation between received samples with sample distance  $k$ . The function of shift register is to store the last fraction of received samples in the previous buffer for processing in the current buffer thus avoid missing detection. The subsequent node calculates the argument in (17). Note, the argument can be interpreted as the convolution between autocorrelation vectors of received samples and of reference samples (one of them should be reversed). Thus, the argument of (17) can be computed more efficiently by fast fourier transform (FFT).

The middle level of the dashed area in Figure 13 calculates the numerator of phasor estimate for SD estimator in (9). Compared with the expression of argument in (17), the numerator of (9) performs a time-varying convolution, which cannot

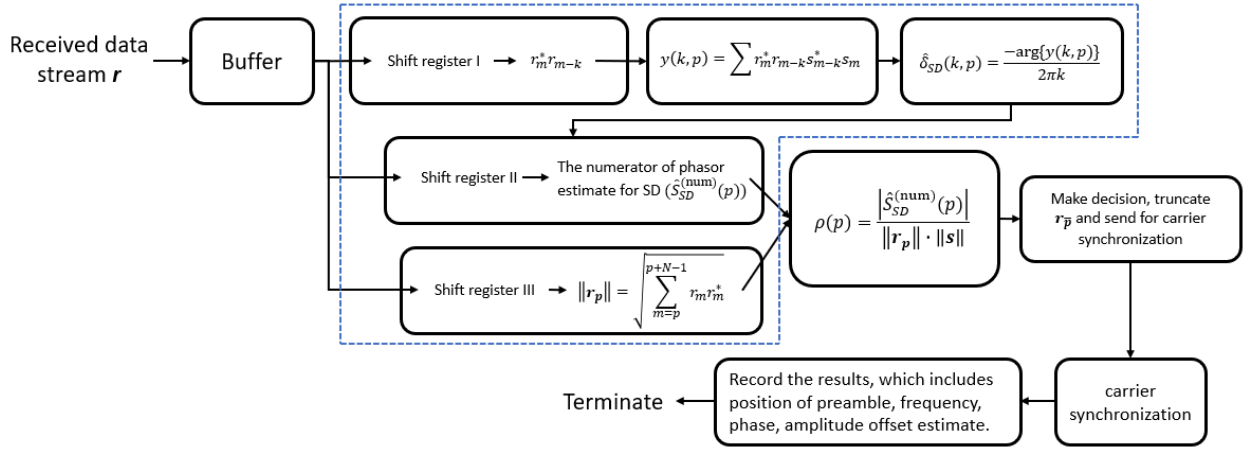


Fig. 13. Block diagram for implementing the proposed joint detection and estimation algorithm in Threading Building Blocks

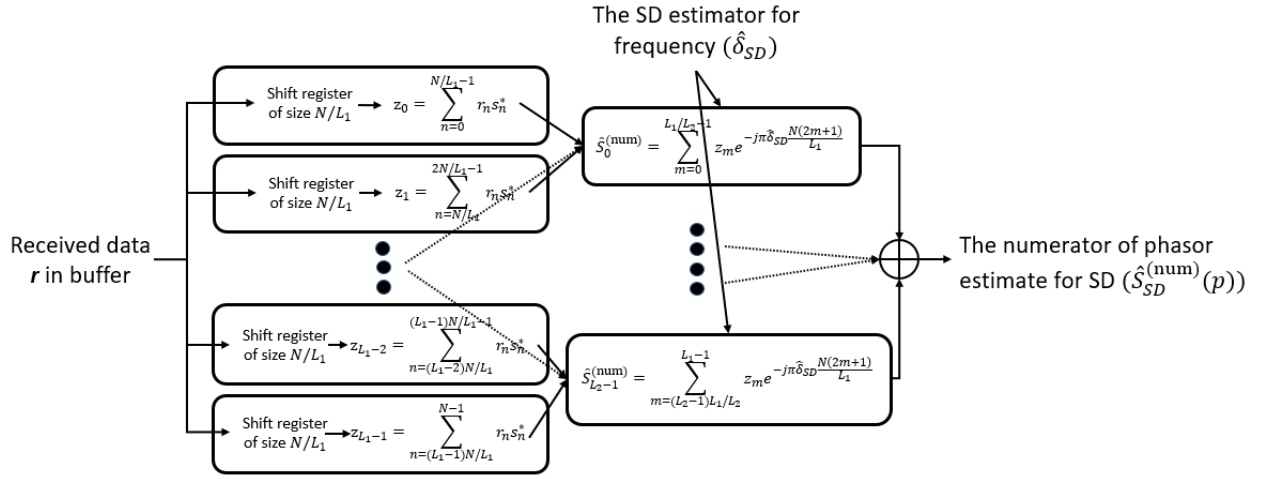


Fig. 14. Block diagram for complexity-simplified computation of phasor estimator (the middle level of dashed area in Figure 13)

be computed by FFT. However, we can still simplify the computation by applying functional parallelism in flow graph. Figure 14 illustrates the details of improving the computation in (9).  $\hat{s}^{(num)}$  is now computed in two stages: The inner stage has  $L_1$  parallel sub nodes each with a shift register in the size of  $N/L_1$ ; the  $L_1$  number of nodes equally separate the received sequence and calculate the partial dot product between received samples and the preamble. The outer stage has  $L_2$  parallel functional nodes, each corrects the number of  $L_1/L_2$  results from inner stage by frequency estimate at the middle position of the corresponding vector in the inner stage. Then,  $\hat{s}^{(num)}$  is approximated by adding all the results from outer stages, which yields

$$\hat{s}^{(num)} \approx \sum_{i=0}^{L_2-1} \sum_{m=iL_1/L_2}^{(i+1)L_1/L_2-1} e^{-j\pi \hat{\delta} \frac{N(2m+1)}{L_1}} \sum_{n=mN/L_1}^{(m+1)N/L_1-1} r_n s_n^*. \quad (32)$$

The accuracy of (32) depends on the number of  $L_1$  stages. On the other hand,  $L_1$  also determines the number of threads of

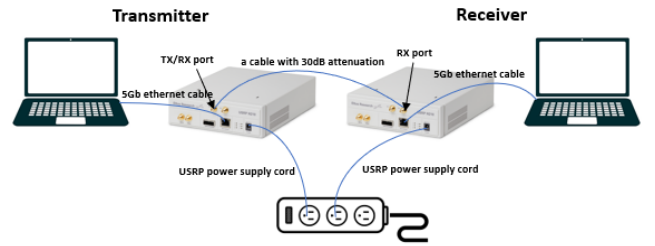


Fig. 15. Machine connection for signal transmission from the transmitter to receiver

Central processing unit (CPU) that will be used when running the program. Thus, the value of  $L_1$  should be determined reasonably. Moreover, to get a good approximation of  $\hat{s}^{(num)}$ , the value of  $L_2$  should be the factor of  $L_1$ . In conclusion, the combination of the two stages lets the computation of  $\hat{s}^{(num)}$  achieving an approximate of  $L_1 \cdot L_2$  speed up.

TABLE I  
BENCHMARK RESULTS OF FUNCTION NODES IN FIGURE 13 (TOP) AND 14 (BOTTOM)

Node name	Time(ns)	CPU(ns)	Iterations
Buffer	408721	407754	1703
AutoC_rx (dashed, top left 1)	160069	160054	3416
C_ctxcrx (dashed, top left 2)	498967	498876	1471
Freq_SD (dashed, top right 1)	187135	187121	3602
Norm_rx (dashed, bottom)	203907	203892	3416
Rho_p (after dashed area, 1th)	837253	837048	829
Dect (after dashed area, 2nd)	378793	378765	1844
Carrier_sync (the last)	811747	811739	844
C_rxtx (left column)	403201	403171	1736
Sub_phasor (right column)	780620	780523	886

### B. Signal Transmission Path

In this section, we briefly talk about the signal transmitting path, which is built for testing our algorithm. The connection of machines is fairly easy: Each of two processors (computers) connected to a universal software radio peripheral (USRP) by one 5-Gigabit Ethernet cable as the transmitter or receiver; Between two USRPs, a cable with 30 dB attenuation connects the RX/TX port and RX port. At the transmitter side, the processor needs to tell the (transmitter) USRP the sample rate, the baseband signal and frequency, the carrier frequency, the transmitter gain, etc. Then, the USRP transmits the analog signal to the (receiver) USRP through the two ports. At the receiver side, the received analog RF signal is first down-converted to baseband, down-sampled to discrete-time data stream and finally stored in the local network. After that, our modified algorithm is going to work by requesting the data from the local network. The outward appearance of the machine connection is shown in Figure 15.

### C. Performance of modified algorithm in TBB

To measure the performance of modified algorithm in TBB, we need to consider three factors comprehensively: accuracy, throughput and latency. Thus, some parameters should be chosen by the following rules. First, the length of preamble should be chosen short to get the largest throughput without degrading the detection and estimation accuracy. Second, the buffer size should be chosen as the power of 2 to achieve the best performance of FFT; Moreover, as discussed, the buffer size should also be chosen to balance between small overhead and short latency. Third, the value of two stages  $L_1$  and  $L_2$  in phasor estimate should be chosen large to best obtain the functional parallelism of flow graph.

As tested, such parameters are determined that can achieve a relatively good performance: The preamble is chosen with the number of symbols  $L_0 = 64$  and oversampling factor  $M = 4$ . The buffer size is 8192. The value of two stages are determined  $L_1 = 2L_2 = 16$ . The specifications of processor that is used for testing the algorithm at the receiver side are given as follows: The CPU is Intel(R) Core(TM) i7-9750H, which includes 6 cores and 12 threads. The best clock speed of CPU can go up to 4500 MHz. Based on above, the google benchmark [20] for

each function node in Figure 13 are shown in Table I. Note, the time cost in the table should refer to one buffer size, i.e., 8192 received samples; Moreover, the throughput of the pipeline is determined by the node with the longest time. Thus, the ideal throughput of the modified algorithm can be approximated by  $8192/837253 \cdot 10^3 \approx 9.78$  MHz.

Based on the above discussion, we test the modified joint detection and estimation algorithm by setting the sample rate at transmitter be 10 MS/s; Moreover, the transmitter gain plus receiver gain is 20 dB. As a result, the maximum throughput of the modified algorithm is in the range of 4.5 MS/s  $\sim$  5.0 MS/s with latency near 1 ms; The detection algorithm is very robust that the false alarm probability is near 0.

### APPENDIX A

#### DERIVATION OF DISTRIBUTION FOR THE SD ESTIMATOR, PROOF OF (21)

To prove the distribution of  $\arg\{W(k)\}$ , where  $W(k)$  is a complex Gaussian random variable, we assume  $W = X + jY$ ,  $X$  and  $Y$  are two Gaussian random variables with distributions  $X \sim \mathcal{N}(\mu_x, \sigma^2)$  and  $Y \sim \mathcal{N}(\mu_y, \sigma^2)$ . Here,  $W$ ,  $X$ ,  $Y$ ,  $\mu_x$ ,  $\mu_y$  and  $\sigma^2$  all depends on  $k$ , we just write those for notation simplicity. We further assume  $\mu_w$  to be the mean of  $W$ . The probability density function of  $W$  is given by

$$f_W(w) = f_{X,Y}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x - \mu_x)^2 + (y - \mu_y)^2}{2\sigma^2}\right). \quad (33)$$

Let  $x = r \cos \theta$ ,  $y = r \sin \theta$ , (33) can be transformed into polar coordinate,

$$\begin{aligned} f_W(w) &= f_{R,\Theta}(r, \theta) \\ &= \frac{r}{2\pi\sigma^2} \exp\left(-\frac{(r \cos \theta - \mu_x)^2 + (r \sin \theta - \mu_y)^2}{2\sigma^2}\right) \\ &= \frac{r}{2\pi\sigma^2} \exp\left(-\frac{r^2 + \mu_x^2 + \mu_y^2}{2\sigma^2}\right) \exp\left(\frac{r}{\sigma^2}(\mu_x \cos \theta + \mu_y \sin \theta)\right) \end{aligned} \quad (34)$$

Plugging  $\mu_x = |\mu_w| \cos(\angle \mu_w)$ ,  $\mu_y = |\mu_w| \sin(\angle \mu_w)$  yields

$$\begin{aligned} f_{R,\Theta}(r, \theta) &= \frac{r}{2\pi\sigma^2} \exp\left(-\frac{r^2 + |\mu_w|^2}{2\sigma^2}\right) \exp\left(\frac{r|\mu_w|}{\sigma^2} \cos(\theta - \angle \mu_w)\right). \end{aligned} \quad (35)$$

Note that  $\theta = \arg\{W(k)\}$ . Thus, we turn our attention to marginal PDF of  $\theta$ ,

$$\begin{aligned}
f_{\Theta}(\theta) &= \int_0^{\infty} \frac{r}{2\pi\sigma^2} \exp\left(-\frac{r^2-2r|\mu_w|\cos(\theta-\angle\mu_w)+|\mu_w|^2}{2\sigma^2}\right) dr \\
&= \int_0^{\infty} \frac{r}{2\pi\sigma^2} \exp\left(-\frac{(r-|\mu_w|\cos(\theta-\angle\mu_w))^2+|\mu_w|^2(1-\cos^2(\theta-\angle\mu_w))}{2\sigma^2}\right) dr \\
&= \frac{1}{2\pi} \exp\left(-\frac{|\mu_w|^2(1-\cos^2(\theta-\angle\mu_w))}{2\sigma^2}\right) \int_0^{\infty} \frac{r}{\sigma^2} \exp\left(-\frac{(r-|\mu_w|\cos(\theta-\angle\mu_w))^2}{2\sigma^2}\right) dr.
\end{aligned} \tag{36}$$

By assuming

$$\begin{aligned}
\alpha &= |\mu_w| \sin(\theta - \angle\mu_w) \\
\beta &= |\mu_w| \cos(\theta - \angle\mu_w) \\
u &= \frac{r - |\mu_w| \cos(\theta - \angle\mu_w)}{\sigma},
\end{aligned}$$

(36) can be simplified as

$$\begin{aligned}
f_{\Theta}(\theta) &= \frac{1}{2\pi} \exp\left(-\frac{\alpha^2}{2\sigma^2}\right) \int_{-\frac{\beta}{\alpha}}^{\infty} \left(u + \frac{\beta}{\alpha}\right) \exp\left(-\frac{u^2}{2}\right) du \\
&= \frac{1}{2\pi} \exp\left(-\frac{\alpha^2}{2\sigma^2}\right) \left( \int_{-\frac{\beta}{\alpha}}^{\infty} u e^{-\frac{u^2}{2}} du + \int_{-\frac{\beta}{\alpha}}^{\infty} \frac{\beta}{\alpha} e^{-\frac{u^2}{2}} du \right) \\
&= \frac{1}{2\pi} \exp\left(-\frac{\alpha^2+\beta^2}{2\sigma^2}\right) + \frac{\beta}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\alpha^2}{2\sigma^2}\right) Q\left(-\frac{\beta}{\alpha}\right) \\
&= \frac{1}{2\pi} \exp\left(-\frac{|\mu_w|^2}{2\sigma^2}\right) + \frac{|\mu_w| \cos(\theta - \angle\mu_w)}{\sqrt{2\pi}\sigma} \exp\left(-\frac{|\mu_w|^2 \sin^2(\theta - \angle\mu_w)}{2\sigma^2}\right) Q\left(-\frac{|\mu_w|}{\sigma} \cos(\theta - \angle\mu_w)\right)
\end{aligned} \tag{37}$$

where  $Q(\cdot)$  is the Q function and (37) gives the explicit marginal PDF for  $\theta$ . Note that  $|\mu_w| = (N-k)\left(\frac{E_s}{T}\right)^2$  from (19). Thus, at relatively high SNR, i.e.,  $|\mu_w| \gg \sigma$ , (37) can be approximated by

$$\begin{aligned}
f_{\Theta}(\theta) &\approx \frac{1}{2\pi} \exp\left(-\frac{|\mu_w|^2}{2\sigma^2}\right) \\
&+ \frac{|\mu_w| \cos(\theta - \angle\mu_w)}{\sqrt{2\pi}\sigma} \exp\left(-\frac{|\mu_w|^2 \sin^2(\theta - \angle\mu_w)}{2\sigma^2}\right) \\
&\left(1 - \frac{\sigma}{\sqrt{2\pi}|\mu_w| \cos(\theta - \angle\mu_w)} \exp\left(-\frac{|\mu_w|^2 \cos^2(\theta - \angle\mu_w)}{2\sigma^2}\right)\right) \\
&= \frac{|\mu_w| \cos(\theta - \angle\mu_w)}{\sqrt{2\pi}\sigma} \exp\left(-\frac{|\mu_w|^2 \sin^2(\theta - \angle\mu_w)}{2\sigma^2}\right)
\end{aligned} \tag{38}$$

The approximation holds because of the property of Q function  $Q(x) \approx \frac{1}{\sqrt{2\pi}x} \exp(-\frac{x^2}{2})$  for  $x \gg 0$ . Then, we only need to look at  $f_{\Theta}(\theta)$  where  $\theta \approx \angle\mu_w$ , i.e.,

$$f_{\Theta}(\theta) \approx \frac{|\mu_w|}{\sqrt{2\pi}\sigma} \exp\left(-\frac{|\mu_w|^2}{2\sigma^2}(\theta - \angle\mu_w)^2\right) \tag{39}$$

Thus,  $\theta$  is approximately Gaussian with the distribution  $\theta \sim \mathcal{N}(\angle\mu_w, \frac{\sigma^2}{|\mu_w|^2})$ , which is equivalent to (21).

## APPENDIX B

### DERIVATION OF THE CRVB FOR FREQUENCY AND PHASE ESTIMATE, PROOF OF (30) AND (31)

In order to derive the CRVB for joint estimation of phase and frequency offset, We define a parameter vector  $\theta \triangleq [\delta \ \phi]$  to include the two parameters that we want to estimate. The first step towards the derivation of the CRVB is to compute the Fisher Information Matrix (FIM,  $\mathbf{I}(\theta)$ ). The calculation of  $\mathbf{I}(\theta)$  is based on log-likelihood function ( $\ln \Lambda$ ), which is carried out in [21, Ch. 4]

$$\ln \Lambda[r(t), \theta] = \frac{2}{N_0} \int_0^{T_0} r(t) s'^*(t, \theta) dt - \frac{1}{N_0} \int_0^{T_0} |s'(t, \theta)|^2 dt \tag{40}$$

where

$$s'(t, \theta) = A e^{j(2\pi\delta t + \phi)} \sum_{i=0}^{L_0-1} c_i g(t - iT), \tag{41}$$

and  $T_0$  is the observation length. Taking the second derivative with respect to each element of the parameter vector  $\theta_i, \theta_j$  yields

$$\begin{aligned}
\frac{\partial^2 \ln \Lambda}{\partial \theta_i \partial \theta_j} &= \frac{2}{N_0} \int_0^{T_0} r(t) \frac{\partial^2 s'^*(t, \theta)}{\partial \theta_i \partial \theta_j} dt - \frac{2}{N_0} \int_0^{T_0} \frac{\partial s'^*(t, \theta)}{\partial \theta_i} \frac{\partial s'(t, \theta)}{\partial \theta_j} dt \\
&- \frac{2}{N_0} \int_0^{T_0} s'(t, \theta) \frac{\partial^2 s'^*(t, \theta)}{\partial \theta_i \partial \theta_j} dt.
\end{aligned} \tag{42}$$

Taking the negative expectation of (42), the elements of FIM are given by

$$\mathbf{I}(\theta)_{ij} = -\mathbb{E} \left[ \frac{\partial^2 \ln \Lambda}{\partial \theta_i \partial \theta_j} \right] = \frac{2}{N_0} \int_0^{T_0} \frac{\partial s'^*(t, \theta)}{\partial \theta_i} \frac{\partial s'(t, \theta)}{\partial \theta_j} dt. \tag{43}$$

By plugging  $s'(t, \theta)$  from (41) into (43) and replacing  $\theta_i, \theta_j$  with  $\delta, \phi$ , the first element of FIM ( $\mathbf{I}(\theta)_{11}$ ) yields

$$\begin{aligned}
\mathbf{I}(\theta)_{11} &= \frac{2}{N_0} \int_0^{T_0} \frac{\partial A e^{j(2\pi\delta t + \phi)} \sum_{i=0}^{L_0-1} c_i g(t - iT)}{\partial \delta} \\
&\frac{\partial A e^{-j(2\pi\delta t + \phi)} \sum_{i=0}^{L_0-1} c_i^* g^*(t - iT)}{\partial \delta} dt \\
&= \frac{2}{N_0} \int_0^{T_0} A^2 4\pi^2 t^2 \left| \sum_{i=0}^{L_0-1} c_i g(t - iT) \right|^2 dt.
\end{aligned} \tag{44}$$

Note that the averaged symbol energy of transmitted signal is calculated by

$$\begin{aligned} E_s &= \int_0^T \left| A \sum_{i=0}^{L_0-1} c_i g(t-iT) \right|^2 dt \\ &\approx \sum_{k=0}^{M-1} \left| A \sum_{i=0}^{L_0-1} c_i g(kT_s - iT) \right|^2 T_s \\ &\approx T \left| A \sum_{i=0}^{L_0-1} c_i g(t-iT) \right|^2, \end{aligned} \quad (45)$$

or,

$$A^2 \left| \sum_{i=0}^{L_0-1} c_i g(t-iT) \right|^2 \approx \frac{E_s}{T}. \quad (46)$$

where  $M = \text{int}(T/T_s)$ . The first approximation in (45) is based on Riemann sum theory.  $\mathbf{I}(\theta)_{11}$  finally results in

$$\mathbf{I}(\theta)_{11} = \frac{8\pi E_s T_0^2 L_0}{3N_0}. \quad (47)$$

Similarly,  $\mathbf{I}(\theta)_{12}$  can be calculated by plugging  $s'(t, \theta)$  from (41) into (43) and replacing  $\theta_i, \theta_j$  with  $\delta, \phi$ , which is given by

$$\begin{aligned} \mathbf{I}(\theta)_{12} &= \frac{2}{N_0} \int_0^{T_0} \frac{\partial A e^{j(2\pi\delta t + \phi)} \sum_{i=0}^{L_0-1} c_i g(t-iT)}{\partial \delta} \\ &\quad \frac{\partial A e^{-j(2\pi\delta t + \phi)} \sum_{i=0}^{L_0-1} c_i^* g^*(t-iT)}{\partial \phi} dt. \end{aligned} \quad (48)$$

Following the same steps as deriving  $\mathbf{I}(\theta)_{11}$ ,  $\mathbf{I}(\theta)_{12}$  can be finally reduced to

$$\mathbf{I}(\theta)_{12} = \frac{2\pi E_s T_0 L_0}{N_0}. \quad (49)$$

$\mathbf{I}(\theta)_{22}$  can be calculated by plugging  $s'(t, \theta)$  from (41) into (43) and replacing  $\theta_i, \theta_j$  with  $\phi, \phi$ , which is given by

$$\begin{aligned} \mathbf{I}(\theta)_{22} &= \frac{2}{N_0} \int_0^{T_0} \frac{\partial A e^{j(2\pi\delta t + \phi)} \sum_{i=0}^{L_0-1} c_i g(t-iT)}{\partial \phi} \\ &\quad \frac{\partial A e^{-j(2\pi\delta t + \phi)} \sum_{i=0}^{L_0-1} c_i^* g^*(t-iT)}{\partial \phi} dt. \end{aligned} \quad (50)$$

$\mathbf{I}(\theta)_{22}$  can be finally reduced to

$$\mathbf{I}(\theta)_{22} = \frac{2E_s L_0}{N_0}. \quad (51)$$

Then,  $\mathbf{I}(\theta)_{11}$ ,  $\mathbf{I}(\theta)_{12}$  and  $\mathbf{I}(\theta)_{22}$  form the FIM

$$\mathbf{I}(\theta) = \begin{bmatrix} \frac{8\pi E_s T_0^2 L_0}{3N_0} & \frac{2\pi E_s T_0 L_0}{N_0} \\ \frac{2\pi E_s T_0 L_0}{N_0} & \frac{2E_s L_0}{N_0} \end{bmatrix}, \quad (52)$$

and the inverse FIM is given by

$$\mathbf{I}^{-1}(\theta) = \begin{bmatrix} \frac{3}{2\pi^2 L_0 T_0^2 E_s / N_0} & \frac{-3}{2\pi L_0 T_0 E_s / N_0} \\ \frac{-3}{2\pi L_0 T_0 E_s / N_0} & \frac{2}{L_0 E_s / N_0} \end{bmatrix}. \quad (53)$$

Thus, the CRVB for the frequency and phase estimates are

$$\begin{aligned} \text{CRVB}(\delta) &\geq \frac{3}{2\pi^2 L_0 T_0^2 E_s / N_0} \\ \text{CRVB}(\phi) &\geq \frac{2}{L_0 E_s / N_0}, \end{aligned} \quad (54)$$

which are equivalent to (30) and (31) respectively by replacing  $T_0$  with  $N$ .

## REFERENCES

- [1] D. D. Falconer, F. Adachi, and B. Gudmundson, "Time division multiple access methods for wireless personal communications," *IEEE Communications Magazine*, vol. 33, no. 1, pp. 50–57, 1995.
- [2] M. Morelli and U. Mengali, "Feedforward frequency estimation for PSK: A tutorial review," *European Transactions on Telecommunications*, vol. 9, pp. 103–116, 1998.
- [3] D. Rife and R. Boorstyn, "Single tone parameter estimation from discrete-time observations," *IEEE Transactions on Information Theory*, vol. 20, no. 5, pp. 591–598, 1974.
- [4] S. Tretter, "Estimating the frequency of a noisy sinusoid by linear regression (corresp.)," *IEEE Transactions on Information Theory*, vol. 31, no. 6, pp. 832–835, 1985.
- [5] S. Kay, "A fast and accurate single frequency estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 1987–1990, 1989.
- [6] M. Luise and R. Reggiannini, "Carrier frequency recovery in all-digital modems for burst-mode transmissions," *IEEE Transactions on Communications*, vol. 43, no. 2/3/4, pp. 1169–1178, 1995.
- [7] M. P. Fitz, "Further results in the fast estimation of a single frequency," *IEEE Transactions on Communications*, vol. 42, no. 234, pp. 862–864, 1994.
- [8] U. Mengali and M. Morelli, "Data-aided frequency estimation for burst digital transmission," *IEEE Transactions on Communications*, vol. 45, no. 1, pp. 23–25, 1997.
- [9] P. Kumari, N. Gonzalez-Prelcic, and R. W. Heath, "Investigating the IEEE 802.11ad standard for millimeter wave automotive radar," in *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*, pp. 1–5, 2015.
- [10] W. Liu, T. Wei, Y. Huang, C. Chan, and S. Jou, "All-digital synchronization for SC-OFDM mode of IEEE 802.15.3c and IEEE 802.11ad," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 2, pp. 545–553, 2015.
- [11] E. Grossi, M. Lops, L. Venturino, and A. Zappone, "Opportunistic radar in IEEE 802.11ad networks," *IEEE Transactions on Signal Processing*, vol. 66, no. 9, pp. 2441–2454, 2018.
- [12] S. K. Bolisetti, K. Ahmed, M. Patwary, and M. Abdel-Maguid, "Compressive parametric GLRT detector for airborne MIMO radar," in *2011 International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 1–5, 2011.
- [13] Guolong Cui, Lingjiang Kong, Xiaobo Yang, and Jianyu Yang, "Coincidence of the rao test, wald test, and GLRT of MIMO radar in gaussian clutter," in *2009 IET International Radar Conference*, pp. 1–4, 2009.
- [14] A. Kumar, S. Dwivedi, and A. K. Jagannatham, "GLRT-based spectrum sensing for MIMO SC-FDMA cognitive radio systems in the presence of synchronization impairments," *IEEE Wireless Communications Letters*, vol. 5, no. 3, pp. 280–283, 2016.
- [15] J. Massey, "Optimum frame synchronization," *IEEE Transactions on Communications*, vol. 20, no. 2, pp. 115–119, 1972.
- [16] Gee Lui and H. Tan, "Frame synchronization for direct-detection optical communication systems," *IEEE Transactions on Communications*, vol. 34, no. 3, pp. 227–237, 1986.
- [17] R. Scholtz, "Frame synchronization techniques," *IEEE Transactions on Communications*, vol. 28, no. 8, pp. 1204–1213, 1980.
- [18] Oneapi-src, "Oneapi-src/onetbb: Oneapi threading building blocks (onetbb)." *GitHub*. [Online]. Available: <https://github.com/oneapi-src/onetbb>. [Accessed: 28-Jan-2022].
- [19] J. R. Michael Voss, Rafael Asenjo, *Pro TBB: C++ Parallel Programming with Threading Building Blocks*. 2019.
- [20] Google, "google/benchmark: A microbenchmark support library." *GitHub*. [Online]. Available: <https://github.com/google/benchmark>. [Accessed: 28-Jan-2022].



- [21] H. L. Van Trees, *Detection, estimation, and modulation theory*. New York: Wiley, 1968.