A Low-Complexity Zak-OTFS Receiver with Over-sampling in the Delay-Doppler Domain

Vineetha Yogesh, Sai Pradeep Muppaneni, Sandesh Rao Mattu, and Ananthanarayanan Chockalingam

***Abstract*—In this paper, we present a novel low-complexity receiver that performs detection and channel estimation tasks in Zak-OTFS modulation using over-sampling in the delay-Doppler (DD) domain. Towards this, we partition the over-sampled signal into multiple two-dimensional (2D) blocks, which is equivalent to parallel branch conventional sampling of delay- and Doppler-shifted signals. This partitioning along with processing only a subset of the partitioned 2D blocks aids in reducing complexity. Specifically, in terms of detection, we propose a symbol-wise selection-based detection (SSD) algorithm which makes symbol-wise decisions based on selection among parallel branch outputs. While the partition based processing reduces complexity, the selection-based symbol decisions offer good performance. We also exploit the proposed partitioning for low-complexity path-wise estimation of the DD channel taps based on maximum energy across blocks. Simulation results with fractional DDs show that the proposed detection and channel estimation algorithms for Zak-OTFS achieve good performance at low complexity.**

***Index Terms*—Zak-OTFS modulation, delay-Doppler domain, DD over-sampling, detection/channel estimation.**

I. Introduction

Orthogonal time frequency space (OTFS) modulation is a state-of-the-art modulation that multiplexes information symbols in the delay-Doppler (DD) domain. It effectively handles the high Doppler spreads of the channel in high-mobility environments and achieves robust performance. The multicarrier version of the modulation (called MC-OTFS) introduced in [1] uses an inner multicarrier modulation block for time-frequency (TF) domain to time domain conversion, along with an outer pre-processing block for DD domain to TF domain conversion. MC-OTFS has been shown to perform better than OFDM in high-Doppler channels [1]- [3]. Recently, a new version of OTFS using Zak transform [4], [5], called Zak-OTFS, has been introduced. Zak-OTFS was first introduced in [6], [7], where the authors present the foundational principles and key attributes of Zak-OTFS. Zak-OTFS directly uses the basis functions of the Zak transform for modulation. It has the advantages of a formal mathematical framework using Zak theory to describe the waveform [6], [7].

Detection and channel estimation are two crucial tasks in the Zak-OTFS receiver. Minimum mean square error (MMSE) detection has been widely adopted in the Zak-OTFS literature [7]- [9]. Detection based on a local neighborhood search has been investigated in [10]. For the purpose of detection, knowledge of the effective DD channel consisting of the transmit (Tx) filter, the physical channel, and the receive (Rx) filter is needed. This can be obtained using a model-dependent approach or a model-free approach [7]. In model-dependent approach, the parameters of the physical channel (i.e., delays, Dopplers, and gains of the paths in the physical channel) are estimated using a channel estimation scheme and the estimated parameters are used to construct the effective channel for detection. On the other hand, in model-free approach, the effective channel is obtained by reading off the pilot response in the DD domain at the receiver [7], without the need to separately estimate the physical channel parameters. Our focus in this paper is on an over-sampling based low-complexity framework for detection and channel estimation in Zak-OTFS. The motivation for the over-sampling framework is presented below.

Inspired by the pioneering work in [6], [7], more research on Zak-OTFS have been reported in recent literature [8]- [15]. In [11], the authors study the practical implementations of Zak-OTFS transceiver, where they realize the DD domain twisted convolution through equivalent time domain and frequency domain windowing, thus enabling DD domain signal processing. The work in [12] addresses the question of optimal Zak-OTFS receiver for a doubly-selective channel. It shows that a Rx TC filter that is matched to the twisted convolution of the physical channel with the Tx TC filter, sampled at the DD grid points, is the optimal Zak-OTFS receiver that recovers sufficient statistics for maximum likelihood (ML) detection of the data symbols. Talking about the implementation challenges of this optimal receiver (in Sec. VIII.D), it points out that when the channel path DD parameters are integer valued and align with the DD transmission grid, the sufficient statistics can be computed from the samples taken on the integer points, and fractional samples are not required. It further notes that when the path parameters are non-integer valued (i.e., fractional DD values), samples taken between the grid points are required to recover the sufficient statistics under the time-frequency based TC filter implementation, and that finer sampling between the grid points (i.e., over-sampling) is required to get accurate measurements of output sample values corresponding to path locations, which can potentially yield more capacity. However, this increases the receiver complexity. The above observations provide a motivation to devise low-complexity algorithms for Zak-OTFS receiver under the over-sampling framework[1].

Vineetha Yogesh, Sai Pradeep Muppaneni, and Ananthanarayanan Chockalingam are with the Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore 560012, India. E-mail: {vineethay; saimuppaneni; achockal}@iisc.ac.in.
Sandesh Rao Mattu is currently with the Duke Quantum Center, Duke University, Durham, NC 27708, USA. Email: sandesh.mattu@duke.edu.

[1]Further, in Appendix A, we present an analysis that provides a theoretical insight on the motivation/rationale for over-sampling, through a comparison between the performance of conventional sampling and over-sampling using a minimum distance measure and pairwise error probability (PEP).

Receiver time domain over-sampling in OTFS was first studied in [16] in the context of for MC-OTFS, where a modified message passing detection was proposed for channels with fractional DDs. They demonstrated significant performance improvement over a conventional sampling receiver. Avoiding joint processing of the full over-sampled vector for complexity reasons, the later work in [17] proposed a low-complexity over-sampling receiver based on maximum ratio combining for MC-OTFS.

In this paper, we consider DD domain over-sampling in the case of Zak-OTFS, which has not been reported so far (e.g., Zak-OTFS work in [6]- [15] consider conventional DD sampling on the information grid). The key novelties in our present work are two-fold.

- First, we analytically show that over-sampling offers improved minimum distance and hence better bit error rate (BER) performance compared to conventional sampling (See Appendix A).
- Second, recognizing the issue of increased complexity in direct over-sampling, we 1) propose a novel over-sampling approach which performs conventional sampling on multiple parallel branches, where each branch samples a different delay- and Doppler-shifted version of the received DD signal (see Sec. III-B), and 2) devise low-complexity algorithms for processing the resulting parallel branch samples for channel estimation and signal detection (see Sec. IV). The proposed over-sampling approach and low-complexity algorithms are shown to offer superior performance compared to conventional sampling (see Sec. V).

The key contributions in the paper can be summarized as follows.

- We propose a novel DD domain over-sampling based receiver for Zak-OTFS, derive the end-to-end DD domain I/O relation.
- To reduce the complexity involved in processing the full directly over-sampled vector, we partition the directly over-sampled vector into multiple two-dimensional (2D) blocks and process only a subset of these blocks. The partitioning has a simple structure which can be viewed as conventional sampling of delay- and Doppler-shifted signals. The same partitioning is exploited to achieve low-complexity detection and channel estimation.
- For detection, we propose a symbol-wise selection-based detection (SSD) algorithm, where symbol-wise decisions are made based on selection among the parallel branch outputs. While processing on the partitions reduces complexity, the selection-based symbol decision in the proposed SSD algorithm offers good performance.
- Exploiting the same over-sampling and partitioning applied on a received pilot frame, we devise a low-complexity algorithm for estimation of the delays, Dopplers, and gains of the DD channel taps based on maximum energy across blocks. We present a performance comparison between the proposed channel estimation scheme and the model-free estimation of I/O relation.
- Simulation results demonstrate good performance and complexity attributes of the proposed over-sampling based detection and channel estimation algorithms.

The rest of the paper is organized as follows. The Zak-OTFS system model is introduced in Sec. II. The end-to-end I/O relation of the over-sampled system and the proposed partitioning are presented in Sec. III. The proposed detection and channel estimation algorithms for the over-sampled receiver are presented in Secs. IV-A and IV-B, respectively. Results and discussions are presented in Sec. V. Conclusions are presented in Sec. VI.

II. Zak-OTFS system model

The block diagram of a Zak-OTFS transceiver is shown in Fig. 1. MN information symbols from a modulation alphabet $\mathbb{A}$ are embedded on a fundamental DD grid defined as $\Lambda_{\text{dd}} = \left\{ (k\Delta\tau, l\Delta\nu) \,\middle|\, k = 0,1,\cdots,M-1, l = 0,1,\cdots,N-1 \right\}$, where $\Delta\tau = \frac{1}{B}$ and $\Delta\nu = \frac{1}{T}$ are the resolutions along the delay and Doppler domain, respectively. $B \approx M\nu_p$ is the bandwidth and $T \approx N\tau_p$ is the frame duration, where τ_p and ν_p are fundamental periods along delay and Doppler axis, such that $\tau_p\nu_p = 1$. The information symbols $x[k,l]$s are encoded as per the following equation to obtain a quasi-periodic extension of the signal in the discrete DD domain:

$$x_{\text{dd}}[k+nM, l+mN] = x[k,l]e^{j2\pi n\frac{l}{N}}, n, m \in \mathbb{Z}. \quad (1)$$

The discrete quasi-periodic signal in (1) is converted into a continuous quasi-periodic DD signal by mounting on a continuous quasi-periodic DD impulse train, as

$$x_{\text{dd}}(\tau,\nu) = \sum_{k,l\in\mathbb{Z}} x_{\text{dd}}[k,l]\delta\left(\tau - k\Delta\tau\right)\delta\left(\nu - l\Delta\nu\right), \quad (2)$$

where $\delta(\cdot)$ denotes dirac delta function. Note that (2) occupies infinite time and bandwidth. In order to restrict the time and bandwidth to $\frac{1}{B}$ and $\frac{1}{T}$, respectively, ink(2) is filtered through the DD domain Tx filter $w_{\text{tx}}(\tau,\nu)$, as [7]

$$x_{\text{dd}}^{\text{w}_{\text{tx}}}(\tau,\nu) = w_{\text{tx}}(\tau,\nu) *_\sigma x_{\text{dd}}(\tau,\nu), \quad (3)$$

where $*_\sigma$ denotes the twisted convolution. Twisted convolution is a mathematical operation that generalizes classical convolution by incorporating a twist factor related to phase shift, rendering the operation non-commutative [19]. Twisted convolution between two functions $a(\tau,\nu)$ and $b(\tau,\nu)$ is defined as

$$a(\tau,\nu) *_\sigma b(\tau,\nu) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} a(\tau',\nu')b(\tau-\tau',\nu-\nu') e^{j2\pi\nu'(\tau-\tau')}d\tau' d\nu'. \quad (4)$$

Also, twisted convolution operation between the quasi-periodized DD signal $x_{\text{dd}}(\tau,\nu)$ and a general DD filter/function $w_{\text{tx}}(\tau,\nu)$ preserves the quasi-periodicity at the output (Appendix 2.G. [19]), i.e., the DD signal $x_{\text{dd}}^{w_{\text{tx}}}(\tau,\nu)$ is quasi-periodic. This quasi-periodicity of $x_{\text{dd}}^{w_{\text{tx}}}(\tau,\nu)$ ensures the existence of the corresponding time domain (TD) signal through inverse Zak transform operation. Accordingly, the TD transmit signal is obtained using inverse Zak transform as

$$s_{\text{td}}(t) = \mathcal{Z}_t^{-1}\left(x_{\text{dd}}^{w_{\text{tx}}}(\tau,\nu)\right) = \sqrt{\tau_p}\int_0^{\nu_p} x_{\text{dd}}^{w_{\text{tx}}}(t,\nu)d\nu. \quad (5)$$

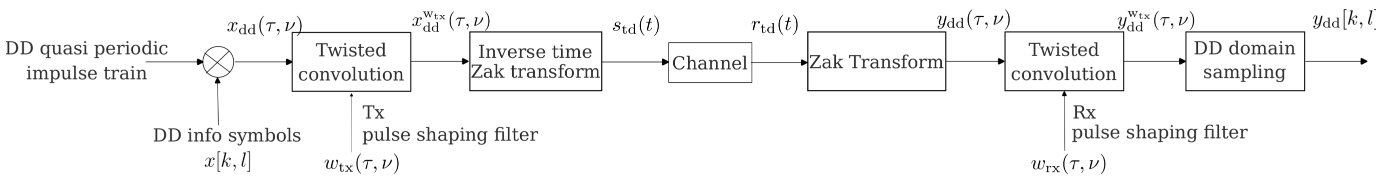


Fig. 1: Block diagram of Zak-OTFS transceiver.

The transmitted signal passes through a doubly-selective channel having P paths whose DD domain impulse response is

$$h(\tau,\nu)=\sum_{i=1}^{P} h_i\delta(\tau-\tau_i)\delta(\nu-\nu_i), \tag{6}$$

where h_i, τ_i, and ν_i are the channel gain, delay, and Doppler of the ith path, respectively. At the receiver, the TD signal is

$$r_{\rm td}(t)=\iint h(\tau,\nu)s_{\rm td}(t-\tau)e^{j2\pi\nu(t-\tau)}d\tau d\nu+n(t), \tag{7}$$

where $n(t)$ is the additive white Gaussian noise (AWGN) with one-sided power spectral density N_0. At the receiver, Zak transform is used to convert the received TD signal $r_{\rm td}(t)$ to DD domain. Zak transform exists for finite energy time domain signals, i.e., any time domain signal in $L^2(\mathbb{R})$ space is mapped to a unique quasi-periodic DD domain signal by Zak transform [5]. The quasi-periodic DD signal at the Zak transform output is given by

$$\begin{aligned} y_{\rm dd}(\tau,\nu) &= \mathcal{Z}_t(r_{\rm td}(t)) \\ &= \sqrt{\tau_p}\sum_{k\in\mathbb{Z}} r_{\rm td}(\tau+k\tau_p)e^{-j2\pi\nu k\tau_p}+n_{\rm dd}(\tau,\nu), \end{aligned} \tag{8}$$

where $n_{\rm dd}(\tau,\nu)=\mathcal{Z}_t(n(t))$ is the noise in DD domain. Next, $y_{\rm dd}(\tau,\nu)$ is filtered through the Rx DD domain filter $w_{\rm rx}(\tau,\nu)$, which is matched to the Tx DD filter, i.e.,

$$w_{\rm rx}(\tau,\nu)=w^*_{\rm tx}(-\tau,-\nu)e^{j2\pi\tau\nu}. \tag{9}$$

The output of the Rx DD filter is given by [7]

$$\begin{aligned} y_{\rm dd}^{w_{\rm rx}}(\tau,\nu) &= w_{\rm rx}(\tau,\nu)*_\sigma y_{\rm dd}(\tau,\nu) \\ &= h_{\rm dd}(\tau,\nu)*_\sigma x_{\rm dd}(\tau,\nu)+n_{\rm dd}^{w_{\rm rx}}(\tau,\nu), \end{aligned} \tag{10}$$

where $h_{\rm dd}(\tau,\nu)$ is the effective continuous DD channel (consisting of the cascade of the Tx DD filter, physical DD channel, and Rx DD filter), given by

$$h_{\rm dd}(\tau,\nu)=w_{\rm rx}(\tau,\nu)*_\sigma h(\tau,\nu)*_\sigma w_{\rm tx}(\tau,\nu), \tag{11}$$

and $n_{\rm dd}^{w_{\rm rx}}(\tau,\nu)=w_{\rm rx}(\tau,\nu)*_\sigma n_{\rm dd}(\tau,\nu)$ is the filtered AWGN in DD domain. Here again, $y_{\rm dd}^{w_{\rm rx}}(\tau,\nu)$ is quasi-periodic due to the property of twisted convolution. The received DD signal $y_{\rm dd}^{w_{\rm rx}}(\tau,\nu)$ is sampled for further receiver processing in the discrete DD domain[2].

Now, let $z(\tau,\nu)$ denote the received information signal in the absence of noise. Using the definition of twisted convolution, $z(\tau,\nu)$ is written as

$$\begin{aligned} z(\tau,\nu) &= h_{\rm dd}(\tau,\nu)*_\sigma x_{\rm dd}(\tau,\nu) \\ &= \iint h_{\rm dd}(\tau',\nu')x_{\rm dd}(\tau-\tau',\nu-\nu')e^{j2\pi\nu'(\tau-\tau')}d\tau'd\nu'. \end{aligned} \tag{12}$$

Substituting (2) in (12), we get

$$\begin{aligned} z(\tau,\nu) &= \sum_{k,l\in\mathbb{Z}} x_{\rm dd}[k,l]h_{\rm dd}(\tau-k\Delta\tau,\nu-l\Delta\nu) \\ &\quad e^{j2\pi(\nu-l\Delta\nu)k\Delta\tau}. \end{aligned} \tag{13}$$

Using the quasi-periodicity in (1) and restricting the DD indices to $k\in\{0,\ldots,M-1\}$ and $l\in\{0,\ldots,N-1\}$ in (13), we obtain

$$z(\tau,\nu)=\sum_{k=0}^{M-1}\sum_{l=0}^{N-1} x_{\rm dd}[k,l]h_{\rm eff}(\tau,\nu,k,l), \tag{14}$$

where $h_{\rm eff}(\tau,\nu,k,l)$ is given by (15) at the top of the next page.

III. Proposed DD domain over-sampling

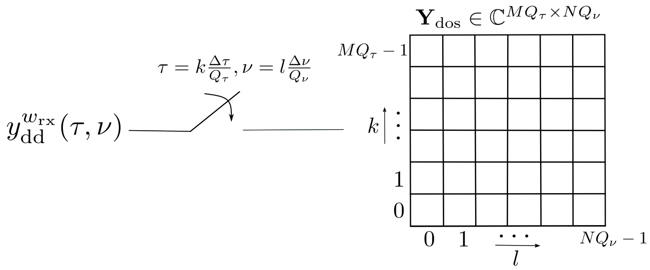
In this section, we consider over-sampling of the continuous DD domain received signal in (10). First, we describe direct over-sampling (DOS). Following this, we describe the proposed parallel branch sampling (PBS) and derive its DD domain I/O relation. The sampling of the signal in (10) involves sampling of the information signal $z(\tau,\nu)=h_{\rm dd}(\tau,\nu)*_\sigma x_{\rm dd}(\tau,\nu)$ and noise $n_{\rm dd}^{w_{\rm rx}}(\tau,\nu)=w_{\rm rx}(\tau,\nu)*_\sigma n_{\rm dd}(\tau,\nu)$.

A. Direct over-sampling

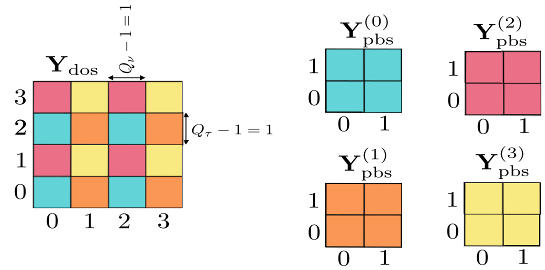
1) Over-sampling of information signal: The signal in (14) is sampled to obtain the discrete DD domain samples. Sampling is done with an over-sampling factor of Q_τ along

[2]It is noted that implementing the continuous DD domain Zak transform on the continuous time domain signal $r_{\rm td}(t)$ $\big($i.e., $y_{\rm dd}(\tau,\nu)=\mathcal{Z}_t(r_{\rm td}(t))\big)$ and the matched filtering operation $y_{\rm dd}^{w_{\rm rx}}(\tau,\nu)=w_{\rm rx}(\tau,\nu)*_\sigma y_{\rm dd}(\tau,\nu)$ in the continuous DD domain at the receiver would require complex analog signal processing. Similar issue arises in the transmitter side as well. Therefore, digital implementation of Zak-OTFS transceiver is of interest. We note that this issue is addressed in Appendix 8.A of [19]. In [19], Appendices 8.A.1 and 8.A.2 present the digital implementations of the Zak-OTFS receiver and transmitter, respectively. Appendix 8.A.1 shows that the receiver output, i.e., $y_{\rm dd}[k,l]=y_{\rm dd}^{\rm rx}\left(\tau=\frac{k\tau_{\rm p}}{M},\nu=\frac{l\nu_{\rm p}}{N}\right)$, can be computed through suitable processing of the discrete time samples of the received continuous time domain signal $r_{\rm td}(t)$. Likewise, Appendix 8.A.2 shows the digital implementation of the Zak-OTFS transmitter.

$$h_{\text{eff}}(\tau,\nu,k,l) = \sum_{m,n\in\mathbb{Z}} h_{\text{dd}}(\tau-(k+nM)\Delta\tau, \nu-(l+mN)\Delta\nu)e^{j2\pi(\nu-(l+mN)\Delta\nu)(k+nM)\Delta\tau}e^{j2\pi\frac{nl}{N}}. \tag{15}$$



(a) DOS and $\mathbf{Y}_{\text{dos}}$ matrix



(b) Partitioning $\mathbf{Y}_{\text{dos}}$ matrix into $\mathbf{Y}_{\text{pbs}}$ matrices for $M = N = 2$, $Q_\tau = Q_\nu = 2$

Fig. 2: DOS in the DD domain and partitioning $\mathbf{Y}_{\text{dos}}$ matrix into multiple $\mathbf{Y}_{\text{pbs}}$ matrices.

the delay axis and Q_ν along the Doppler axis. The sampling resolutions along the delay and Doppler axes are $\frac{\Delta\tau}{Q_\tau}$ and $\frac{\Delta\nu}{Q_\nu}$, respectively, and the fundamental DOS grid is defined as

$$\Lambda_{\text{dos}} = \Big\{\Big(k'\frac{\Delta\tau}{Q_\tau}, l'\frac{\Delta\nu}{Q_\nu}\Big)\Big| \; k'=0,1,\ldots,MQ_\tau-1, l'=0,1,\ldots,NQ_\nu-1\Big\}. \tag{16}$$

The direct over-sampled signal obtained by sampling (14) on Λ_{dos} is given by

$$z_{\text{dos}}[k',l'] = \sum_{k=0}^{M-1}\sum_{l=0}^{N-1} x_{\text{dd}}[k,l] h_{\text{eff}}\left(k'\frac{\Delta\tau}{Q_\tau}, l'\frac{\Delta\nu}{Q_\nu}, k, l\right). \tag{17}$$

2) Over-sampling of noise: While sampling $y_{\text{dd}}^{w_{\text{rx}}}(\tau,\nu)$ given by (10) on Λ_{dos}, the noise $n_{\text{dd}}^{w_{\text{rx}}}(\tau,\nu)$ also gets sampled on Λ_{dos}, resulting in the noise samples $n_{\text{dd}}^{w_{\text{rx}}}[k',l']$, whose covariance statistics affect performance. The $[k_1NQ_\nu+l_1, k_2NQ_\nu+l_2]$th element of noise-covariance matrix $\mathbf{R}_{\text{dos}} \in \mathbb{C}^{MNQ_\tau Q_\nu \times MNQ_\tau Q_\nu}$ is given by

$$\mathbf{R}_{\text{dos}}[k_1NQ_\nu+l_1, k_2NQ_\nu+l_2] = \mathbb{E}[n_{\text{dd}}^{w_{\text{rx}}}[k_1,l_1] n_{\text{dd}}^{w_{\text{rx}}*}[k_2,l_2]], \tag{18}$$

where $k_1,k_2 \in \{0,\ldots,MQ_\tau-1\}$ and $l_1,l_2 \in \{0,\ldots,NQ_\nu-1\}$. The samples of the DOS signal are given by

$$\mathbf{y}_{\text{dos}}[k'NQ_\nu+l'] = z_{\text{dos}}[k',l'] + n_{\text{dd}}^{w_{\text{rx}}}[k',l']. \tag{19}$$

The DD domain I/O relation in matrix-vector form for direct over-sampling can then be written as

$$\mathbf{y}_{\text{dos}} = \mathbf{H}_{\text{dos}}\mathbf{x} + \mathbf{n}_{\text{dos}}, \tag{20}$$

where $\mathbf{y}_{\text{dos}} \in \mathbb{C}^{MNQ_\tau Q_\nu \times 1}, \mathbf{x} \in \mathbb{C}^{MN\times 1}$, $\mathbf{H}_{\text{dos}} \in \mathbb{C}^{MNQ_\tau Q_\nu \times MN}$, and $\mathbf{n}_{\text{dos}} \in \mathbb{C}^{MNQ_\tau Q_\nu \times 1}$, with $\mathbf{x}[kN+l] = x_{\text{dd}}[k,l]$, $\mathbf{H}_{\text{dos}}[k'NQ_\nu+l', kN+l] = h_{\text{eff}}(k'\frac{\Delta\tau}{Q_\tau}, l'\frac{\Delta\nu}{Q_\nu}, k, l)$ given by (21) at the top of the next page, $\mathbf{n}_{\text{dos}}[k'NQ_\nu+l'] = n_{\text{dd}}^{w_{\text{rx}}}[k',l']$, and $\mathbf{y}_{\text{dos}}[k'NQ_\nu+l']$ given by (19). Note that by setting $Q_\tau = Q_\nu = 1$, we get the sampling instants corresponding to the case of conventional sampling, which is sampling on the DD grid Λ_{dd}.

Remark: Using the DOS system model in (20), in Appendix A, we present an analysis that provides a theoretical insight on the motivation/rationale for over-sampling. Our analysis in Appendix A shows that over-sampling provides a larger minimum distance and smaller PEP compared to conventional sampling, and hence can achieve better BER performance (see Fig. 12).

B. Proposed parallel branch sampling

The detection and estimation complexity using the DOS system model in (20) increases as the over-sampling factors Q_τ and Q_ν are increased. Therefore, in order to reap the benefits of over-sampling at low complexity, we propose partitioning the 2D over-sampled signal into blocks of manageable sizes. The partitioning procedure is equivalent to parallel branch conventional sampling of the delay- and Doppler-shifted received signal. We first describe the procedure of partitioning the 2D discrete DD domain received signal obtained using DOS. Following this, we establish the equivalence between the partitioned DD domain blocks and the blocks obtained using parallel branch conventional sampling of the delay- and Doppler-shifted versions of the received signal.

1) Partitioning the DOS signal vector: The DOS signal vector $\mathbf{y}_{\text{dos}}$ of size $MNQ_\tau Q_\nu \times 1$ in (20) is partitioned into $Q_\tau Q_\nu$ 2D blocks/matrices, each of size $M\times N$, as follows. Let $\mathbf{Y}_{\text{pbs}}^{(q_\tau+Q_\tau q_\nu)}$ denote the $(q_\tau+Q_\tau q_\nu)$th partitioned block of size $M\times N$ for a given q_τ and q_ν, where $q_\tau \in \{0,1,\ldots,Q_\tau-1\}$ and $q_\nu \in \{0,1,\ldots,Q_\nu-1\}$. The 2D block $\mathbf{Y}_{\text{pbs}}^{(q_\tau+Q_\tau q_\nu)}$ is obtained from $\mathbf{y}_{\text{dos}}$ as

$$\mathbf{Y}_{\text{pbs}}^{(q_\tau+Q_\tau q_\nu)} = \mathbf{Y}_{\text{dos}}[q_\tau : Q_\tau : MQ_\tau - Q_\tau + q_\tau, \; q_\nu : Q_\nu : NQ_\nu - Q_\nu + q_\nu], \tag{22}$$

where $\mathbf{Y}_{\text{dos}}[k,l] = \mathbf{y}_{\text{dos}}[kNQ_\nu+l]$ and $[a:c:b] = \{a+sc \mid s=0,1,2,\ldots$ and $a+sc \le b\}$. Note that, in the $\mathbf{Y}_{\text{dos}}$ matrix of size $MQ_\tau \times NQ_\nu$, the spacing between two adjacent cells of the $(q_\tau+Q_\tau q_\nu)$th partition along delay and Doppler axis is $(Q_\tau-1)$ and $(Q_\nu-1)$, respectively. Also, each partition in $\mathbf{Y}_{\text{dos}}$ consists of MN cells which constitute one $\mathbf{Y}_{\text{pbs}}$ matrix of size $M\times N$. In Fig. 2, we present an illustration of the proposed partitioning for a system with $M=N=2$ and $Q_\tau = Q_\nu = 2$, where Fig. 2(a) shows the direct over-sampling of $y_{\text{dd}}^{w_{\text{rx}}}(\tau,\nu)$ and the corresponding $MQ_\tau \times NQ_\nu$-sized $\mathbf{Y}_{\text{dos}}$

$$\mathbf{H}_{\text{dos}}[k'NQ_\nu+l', kN+l] = \sum_{m,n\in\mathbb{Z}} h_{\text{dd}}\left(k'\frac{\Delta\tau}{Q_\tau}-(k+nM)\Delta\tau, l'\frac{\Delta\nu}{Q_\nu}-(l+mN)\Delta\nu\right)e^{j2\pi\frac{(l'-(l+mN)Q_\nu)(k+nM)}{MNQ_\nu}}e^{j2\pi\frac{nl}{N}}. \quad (21)$$

matrix (i.e., 4×4 matrix in this case), and Fig. 2(b) shows the partitioning of $\mathbf{Y}_{\text{dos}}$ matrix into $Q_\tau Q_\nu$ number of $\mathbf{Y}_{\text{pbs}}$ matrices each of size $M\times N$ (i.e., 4 matrices, each of size 2×2) as per (22).

2) Equivalence of proposed partitioning to parallel branch conventional sampling: The partitioning described in the previous subsection can be viewed as performing conventional sampling (i.e., $Q_\tau = Q_\nu = 1$) on parallel branches as shown in Fig. 3, where each branch samples a delay- and Doppler-shifted version of the received signal. That is, each branch is sampled at a delay resolution of $\Delta\tau$ and a Doppler resolution of $\Delta\nu$. The input to the $q' = q_\tau + Q_\tau q_\nu$th branch, $q' \in \{0,1,\ldots,Q_\tau Q_\nu - 1\}$ is delayed along delay domain by $\frac{q_\tau}{Q}\Delta\tau$ and along Doppler domain by $\frac{q_\nu}{Q_\nu}\Delta\nu$, i.e., the input signal to the q'th branch is $y_{\text{dd}}^{w_{\text{rx}}}(\tau-\frac{q_\tau}{Q_\tau}\Delta\tau, \nu-\frac{q_\nu}{Q_\nu}\Delta\nu)$. The above is illustrated in Fig. 3 for $Q_\tau = Q_\nu = 2$. The samples from the $Q_\tau Q_\nu$ branches can be processed for symbol detection and channel estimation.

Note that for the q' branch, the sampling resolutions along delay and Doppler axis are $\Delta\tau$ and $\Delta\nu$, respectively. However, the signal that is sampled on q'th branch has a fixed offset of $\frac{q_\tau}{Q_\tau}$ and $\frac{q_\nu}{Q_\nu}$ along delay and Doppler axis, respectively. Thus effectively, the sampling on q'th branch is at $(k'+\frac{q_\tau}{Q_\tau})\Delta\tau$ along delay axis and $(l'+\frac{q_\nu}{Q_\nu})\Delta\nu$ along Doppler axis, where $k' = 0,1,\ldots,M-1$ and $l' = 0,1,\ldots,N-1$. Note that these sampling points are fractional in nature. This fractional nature of the sampling points is exploited while formulating the proposed channel estimation algorithm in Sec. IV-B.

3) Sampling of information signal: The sampled signal obtained from q'th branch by sampling $z(\tau-\frac{q_\tau}{Q_\tau}\Delta\tau, \nu-\frac{q_\nu}{Q_\nu}\Delta\nu)$ (see (14)) on Λ_{dd} is given by

$$z^{(q')}[k',l'] = \sum_{k=0}^{M-1}\sum_{l=0}^{N-1} x_{\text{dd}}[k,l] \; h_{\text{eff}}\left(\left(k'+\frac{q_\tau}{Q_\tau}\right)\Delta\tau, \left(l'+\frac{q_\nu}{Q_\nu}\right)\Delta\nu, k, l\right). \quad (23)$$

4) Sampling of noise: Delaying the input to the q'th branch results in the noise signal $n_{\text{dd}}^{w_{\text{rx}}}(\tau-\frac{q_\tau}{Q_\tau}\Delta\tau, \nu-\frac{q_\nu}{Q_\nu}\Delta\nu)$. For q'th branch, with filtered noise samples $n_{\text{dd}}^{w_{\text{rx}}(q')}[k',l']$, the $[k_1N+l_1, k_2N+l_2]$th element of the noise covariance matrix $\mathbf{R}^{(q')}$ is

$$\mathbf{R}^{(q')}[k_1N+l_1, k_2N+l_2] = \mathbb{E}[n_{\text{dd}}^{w_{\text{rx}}(q')}[k_1,l_1] n_{\text{dd}}^{w_{\text{rx}}(q')*}[k_2,l_2]], \quad (24)$$

where $\mathbf{R}^{(q')} \in \mathbb{C}^{MN\times MN}$, $k_1,k_2 \in \{0,1,\ldots,M-1\}$, and $l_1,l_2 \in \{0,1,\ldots,N-1\}$.

The matrix-vector DD domain I/O relation corresponding to the q'th branch, with parallel branch sampling, is given by

$$\mathbf{y}^{(q')} = \mathbf{H}^{(q')}\mathbf{x} + \mathbf{n}^{(q')}, \quad (25)$$

where $\mathbf{y}^{(q')} \in \mathbb{C}^{MN\times 1}, \mathbf{x} \in \mathbb{C}^{MN\times 1}$, $\mathbf{H}^{(q')} \in \mathbb{C}^{MN\times MN}$, and $\mathbf{n}^{(q')} \in \mathbb{C}^{MN\times 1}$, such that $\mathbf{x}[kN+l] = x_{\text{dd}}[k,l]$, $\mathbf{H}^{(q')}[k'N+l', kN+l] = h_{\text{eff}}((k'+\frac{q_\tau}{Q_\tau})\Delta\tau, (l'+\frac{q_\nu}{Q_\nu})\Delta\nu, k, l)$ given by (26) at the top of the next page, $\mathbf{n}^{(q')}[k'N+l'] = n_{\text{dd}}^{w_{\text{rx}}(q')}[k',l']$, and $\mathbf{y}^{(q')}[k'N+l'] = z^{(q')}[k',l'] + n_{\text{dd}}^{w_{\text{rx}}(q')}[k',l']$. Let $\mathbf{Y}^{(q')}$ denote the $M\times N$ matrix corresponding to the q'th branch constructed such that $\mathbf{Y}^{(q')}[k,l] = \mathbf{y}^{(q')}[kN+l]$. While the detection algorithm proposed in Sec. IV-A uses the $\mathbf{y}^{(q')}$ form of the received signal, the channel estimation algorithm proposed in Sec. IV-B uses the $\mathbf{Y}^{(q')}$ form. Also, note that the covariance matrices for the noise vectors in (20) and (25) for DOS and PBS, respectively, are required for both detection and channel estimation. These covariance matrices for DOS and PBS with sinc filter will be derived in Appendix D.

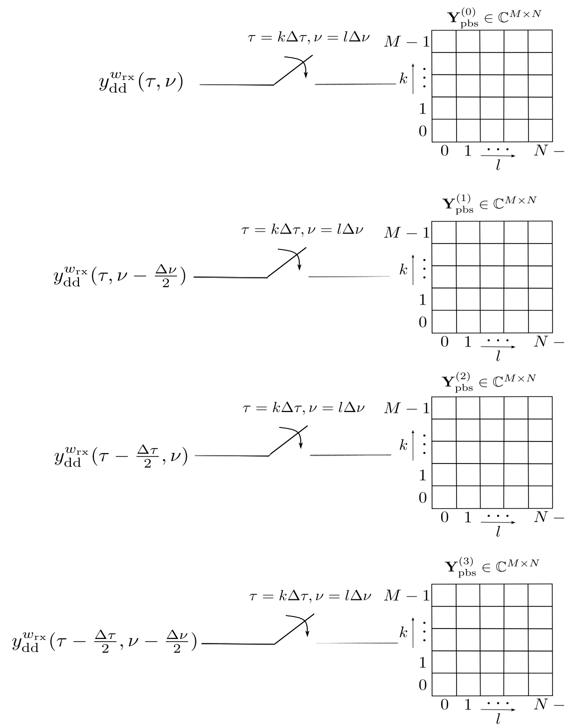


Fig. 3: Parallel branch sampling for $Q_\tau = Q_\nu = 2$.

IV. Proposed algorithms for detection and channel estimation

Using the over-sampled system model developed in the previous section, here, we present the proposed low-complexity algorithms for detection and DD channel estimation.

A. Proposed symbol-wise selection-based detection algorithm

The proposed symbol-wise selection-based detection (SSD) algorithm performs, as described below. For an over-sampling factor of Q_τ along delay and Q_ν along Doppler axes, we have $Q_\tau Q_\nu$ parallel branches or equivalently $Q_\tau Q_\nu$ sampled blocks of size $M\times N$. The DD domain I/O relation corresponding

$$\mathbf{H}^{(q')}[k'N+l',kN+l]=\sum_{m,n\in\mathbb{Z}} h_{\mathrm{dd}}\left(\left(k'+\frac{q_\tau}{Q_\tau}-k-nM\right)\Delta\tau,\left(l'+\frac{q_\nu}{Q_\nu}-l-mN\right)\Delta\nu\right)e^{j2\pi\frac{(l'+\frac{q_\nu}{Q_\nu}-(l+mN))(k+nM)}{MN}}e^{j2\pi\frac{nl}{N}}. \quad (26)$$

to the $q'=(q_\tau+Q_\tau q_\nu)$th branch, $q_\tau=0,\ldots,Q_\tau-1$, $q_\nu=0,\cdots,Q_\nu-1$, is given by (25). Minimum mean square error (MMSE) equalization is considered on each branch. The MMSE equalizer on the q'th branch is given by

$$\mathbf{H}_{\mathrm{eq}}^{(q')}=\mathbf{H}^{(q')H}(\mathbf{H}^{(q')}\mathbf{H}^{(q')H}+\mathbf{R}^{(q')})^{-1}, \quad (27)$$

where $\mathbf{R}^{(q')}$ is the noise covariance matrix corresponding to the q'th branch whose elements are given by (24). The equalized signal on q'th branch is given by

$$\tilde{\mathbf{y}}^{(q')}=\mathbf{H}_{\mathrm{eq}}^{(q')}\mathbf{y}^{(q')}. \quad (28)$$

For each equalized symbol $\tilde{\mathbf{y}}^{(q')}[i], i\in[1,\ldots,MN]$, that symbol from $\mathbb{A}$ having minimum Euclidean distance from $\tilde{\mathbf{y}}^{(q')}[i]$ and the corresponding minimum distance are

$$\mathbf{e}^{(q')}[i] = \arg\min_{a\in\mathbb{A}}\{|\tilde{\mathbf{y}}^{(q')}[i]-a|\}, \quad (29)$$

$$\mathbf{d}^{(q')}[i] = \min_{a\in\mathbb{A}}\{|\tilde{\mathbf{y}}^{(q')}[i]-a|\}, \quad (30)$$

respectively, where $|\cdot|$ denotes the absolute value. Using the vector $\left\{\mathbf{d}^{(q')}\right\}_{q'=0}^{Q_\tau Q_\nu-1}$, $\tilde{q}$ is obtained as

$$\tilde{q}=\arg\min_{q'=0,\cdots,Q_\tau Q_\nu-1}\mathbf{d}^{(q')}[i], \quad (31)$$

and ith symbol in the Zak-OTFS frame is detected as $\hat{\mathbf{x}}[i]=\mathbf{e}^{(\tilde{q})}[i]$. **Algorithm 1** gives the listing of the SSD algorithm described above.

Complexity: The SSD algorithm involves computation of the inverse of $MN\times MN$ matrices, $Q_\tau Q_\nu$ times. So, the computational complexity is $Q_\tau Q_\nu\mathcal{O}(M^3N^3)$. This complexity is less compared to the complexity of joint MMSE detection (JD) carried out on the full DOS vector, which is $\mathcal{O}(M^3N^3Q_\tau^3Q_\nu^3)$.

1) Subset selection among parallel branches: To reduce complexity, we propose to choose and process only a subset of the parallel branches. Specifically, out of the available $Q_\tau Q_\nu$ branches, choose those branches having equal relative shifts along the delay and Doppler domains, i.e., choose those branches for which $q_\tau=q_\nu$[3]. So, the number of selected branches is $\min(Q_\tau,Q_\nu)$. For the sampling illustration in Fig. 3 with $Q_\tau=Q_\nu=2$, for $M=N=2$, the first and fourth branches will be chosen, i.e., the 2D blocks $\mathbf{Y}_{\mathrm{pbs}}^{(0)}$ and $\mathbf{Y}_{\mathrm{pbs}}^{(3)}$ in Fig. 2(b) are chosen for processing and $\mathbf{Y}_{\mathrm{pbs}}^{(1)}$ and $\mathbf{Y}_{\mathrm{pbs}}^{(2)}$ are ignored. In general, this corresponds to choosing $q_\tau(Q_\tau+1)$th branch, $\forall\, q_\tau\in[0,\ldots,Q_\tau-1]$, from the available $Q_\tau Q_\nu$ branches. For detection with subset selection, in line 2 of **Algorithm 1**, only a subset of $Q_\tau Q_\nu-1$ branches, i.e., $q'\in\{0,(Q_\tau+1),2(Q_\tau+1),\ldots,(Q_\tau-1)(Q_\tau+1)\}$, is considered, and the resulting complexity is $\min(Q_\tau,Q_\nu)\mathcal{O}(M^3N^3)$.

Algorithm 1 Proposed SSD algorithm

1: **Inputs:** $\mathbf{y}^{(q')},\mathbf{H}^{(q')},\mathbf{R}^{(q')}$, $\forall\ q'=(q_\tau+Q_\tau q_\nu)$, $q_\tau\in[0,\ldots,Q_\tau-1]$ and $q_\nu\in[0,\ldots,Q_\nu-1]$, $\mathbb{A}$
2: **for** $q'=0$ to $Q_\tau Q_\nu-1$ **do**
3: Obtain $\mathbf{H}_{\mathrm{eq}}^{(q')}=\mathbf{H}^{(q')H}(\mathbf{H}^{(q')}\mathbf{H}^{(q')H}+\mathbf{R}^{(q')})^{-1}$
4: Compute $\tilde{\mathbf{y}}^{(q')}=\mathbf{H}_{\mathrm{eq}}^{(q')}\mathbf{y}^{(q')}$
5: **for** $i=1$ to MN
6: $\mathbf{e}^{(q')}[i]=\arg\min_{a\in\mathbb{A}}\{|\tilde{\mathbf{y}}^{(q')}[i]-a|\}$
7: $\mathbf{d}^{(q')}[i]=\min_{a\in\mathbb{A}}\{|\tilde{\mathbf{y}}^{(q')}[i]-a|\}$
8: **end for**
9: **end for**
10: **for** $i=1$ to MN **do**
11: Obtain $\tilde{q}=\arg\min_{q'=0,\ldots,Q_\tau Q_\nu-1}\mathbf{d}^{(q')}[i]$
12: $\hat{\mathbf{x}}[i]=\mathbf{e}^{(\tilde{q})}[i]$
13: **end for**
14: **Output:** $\hat{\mathbf{x}}$

B. Proposed channel estimation algorithm

The proposed DD channel estimation algorithm that uses the parallel branch outputs is presented here. An exclusive DD pilot frame[4] is transmitted for the purpose of channel estimation. The $M\times N$ pilot matrix, denoted by $\mathbf{X}_{\mathrm{pil}}$, has a pilot symbol at location $(k=k_{\mathrm{pil}},l=l_{\mathrm{pil}})$ and zeros in all other locations. The received pilot signal is subjected to parallel branch sampling at the receiver. The resultant $MN\times 1$ vector at the output of q'th branch, $q'\in\{0,1,\ldots,Q_\tau Q_\nu-1\}$, is denoted by $\mathbf{y}_{\mathrm{pil}}^{(q')}$ (see (25)). The $M\times N$ matrix form of $\mathbf{y}_{\mathrm{pil}}^{(q')}$ is constructed as $\mathbf{Y}_{\mathrm{pil}}^{(q')}[k,l]=\mathbf{y}_{\mathrm{pil}}^{(q')}[kN+l]$. The idea behind the proposed channel estimation algorithm for fractional DD channels is described in the following paragraph.

Consider the case when the path delays and Dopplers are integer multiples of conventional sampling resolutions along delay and Doppler axis, i.e, $\Delta\tau$ and $\Delta\nu$, respectively. Then the received pilot symbol corresponding to ith path is shifted by $\alpha_i=\frac{\tau_i}{\Delta\tau}$ along delay axis and $\beta_i=\frac{\nu_i}{\Delta\nu}$ along Doppler axis with respect to the transmitted pilot location $(k=k_{\mathrm{pil}},l=l_{\mathrm{pil}})$. Hence, estimating α_i and β_i amounts to

[3]*Remark on the choice of subset selection:* We note that the relative shifts along delay and Doppler is 0 for the case of conventional sampling, i.e., $q_\tau=q_\nu=0$. Along these lines, for subset selection, we have chosen symmetric sampling along the delay and Doppler axes with $q_\tau=q_\nu=1,2,...$ for over-sampling. We note that this is a simple choice and that more research is possible to determine the optimum subset selection. Specifically, the proposed choice is one way out of $\binom{Q_\tau Q_\nu}{\min(Q_\tau,Q_\nu)}$ possible ways of choosing $\min(Q_\tau,Q_\nu)$ branches from the available set of $Q_\tau Q_\nu$ branches. More research can be carried out to optimize the minimum number of branches needed and the choice of branches in order to achieve performance close to that with processing all the branches, as future work.

[4]The use of exclusive pilot frame in the current work is aimed at demonstrating the effectiveness of the over-sampling framework for channel estimation and detection in Zak-OTFS. Exclusive pilot frames, though simple, result in throughput loss and high peak to average power ratio (PAPR). Embedded and superimposed pilot frames that can alleviate these issues can be considered in the over-sampling framework as future work.

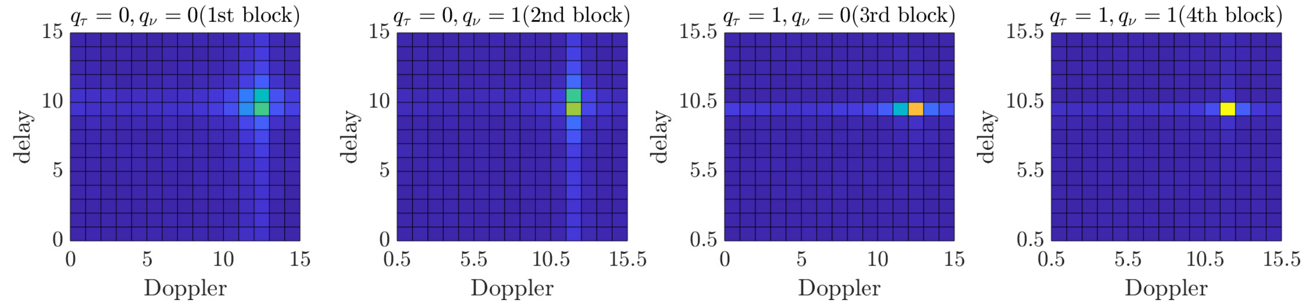


Fig. 4: Response to the pilot as observed at the output of the parallel branches for $M = N = 16$, $k_{\text{pil}} = l_{\text{pil}} = 8$, $P = 1$, $h_1 = 1$, $\alpha_1 = 1.45$, $\beta_1 = 3.65$, $Q_\tau = Q_\nu = 2$, and no noise.

finding the location of the maximum energy bin in the received pilot matrix and calculate its shift along delay and Doppler axis with respect to $(k = k_{\text{pil}}, l = l_{\text{pil}})$. When the DD channel is fractional, i.e., α_i and β_i are fractional multiples of $\Delta\tau$ and $\Delta\nu$, respectively, the procedure described above cannot recover the fractional parts of α_i and β_i with conventional sampling. In order to estimate the fractional DD accurately, it is required that sampling must be done at fractional multiples of $\Delta\tau$ and $\Delta\nu$, i.e., over-sampling along delay and Doppler axis. This is illustrated in Fig. 4, which shows the parallel matrices $\mathbf{Y}_{\text{pil}}^{(q')}$ obtained by transmitting a 16×16 pilot frame with pilot located at $k_{\text{pil}} = l_{\text{pil}} = 8$. One path channel with $h_1 = 1$, $\alpha_1 = 1.45$, $\beta_1 = 3.65$ and an over-sampling factor of $Q_\tau = Q_\nu = 2$ are considered. Recall that the $q' = q_\tau + Q_\tau q_\nu$th branch sampling grid has $\left\{k + \frac{q_\tau}{Q_\tau} \middle| k = 0, \cdots, M-1\right\}$ indices along delay and $\left\{l + \frac{q_\nu}{Q_\nu} \middle| l = 0, \cdots, N-1\right\}$ along Doppler axis. The illustration in Fig. 4 shows that each matrix has peaks in bins corresponding to relative shift of α_1 and β_1 from k_{pil} and l_{pil}, respectively. Note that the localization of the received pilot and the intensity of the peaks depend on q_τ and q_ν. For the block with $q_\tau = q_\nu = 0$ (1st block), the intensity of the peak is minimum among the four blocks with the peak at $(9, 12)$. The corresponding estimates are $(\hat{\alpha}, \hat{\beta}) = (9-8, 12-8) = (1, 4)$, resulting in an absolute error of $|1.45-1| = 0.45$ in the delay index and $|3.65-4| = 0.45$ in the Doppler index. However, for the block with $q_\tau = q_\nu = 1$ (4th block), the intensity of the peak is maximum among the four blocks, with the peak at $(9.5, 11.5)$. The corresponding estimates are $(\hat{\alpha}, \hat{\beta}) = (9.5-8, 11.5-8) = (1.5, 3.5)$, resulting in an absolute error of $(0.05, 0.15)$, which is the least among the errors of the four blocks.

Based on the illustration above, we proceed to devise the channel estimation algorithm using parallel branch sampling. Towards this, we formulate an alternate DD domain input-output relation equation to facilitate channel estimation. In formulating the algorithm, we do not assume separability of paths in the delay or Doppler domain, i.e., paths can interfere. The alternate DD domain input-output equation is given by

$$\mathbf{y}_{\text{pil}}^{(q')} = \mathbf{\Phi}_{\text{pil}}^{(q')} \mathbf{h} + \mathbf{n}_{\text{pil}}^{(q')}, \tag{32}$$

where $\mathbf{\Phi}_{\text{pil}}^{(q')} = [\mathbf{\Psi}_1^{(q')}(\tau_1, \nu_1)\ \mathbf{x}_{\text{pil}}\ \ldots\ \mathbf{\Psi}_P^{(q')}(\tau_p, \nu_p)\ \mathbf{x}_{\text{pil}}] \in \mathbb{C}^{MN \times P}$, $\mathbf{x}_{\text{pil}} \in \mathbb{R}^{MN \times 1}$ such that $\mathbf{x}_{\text{pil}}[kN + l] = \mathbf{X}_{\text{pil}}[k, l]$, $\mathbf{h} = [h_1\ \ldots\ h_P]^T$, where $[\cdot]^T$ denotes transpose operation, and $\mathbf{\Psi}_i^{(q')}(\tau_i, \nu_i) \in \mathbb{C}^{MN \times MN}$ is the effective channel matrix when the channel has only ith path, given by

$$\mathbf{\Psi}_i^{(q')}(\tau_i, \nu_i) = \mathbf{H}^{(q')} \tag{33}$$
$$\text{with } h_i = 1, \tau_u = \nu_u = h_u = 0, \forall u \neq i,\ u \in \{1, \ldots, P\}.$$

As mentioned above, the proposed channel estimation algorithm operates on $\mathbf{Y}_{\text{pil}}^{(q')}$ matrices, $q' \in \{0, 1, \cdots, Q_\tau Q_\nu - 1\}$. It is iterative in nature and progresses path-wise, i.e., one path is estimated and canceled in one iteration. Let j denote the iteration index such that $1 < j \leq P_{\max}$, where $P_{\max} > P$ denotes the maximum number of paths estimated after which the algorithm terminates. The jth iteration comprises two major operations. First, estimate the delay, Doppler, and channel gain of the jth path. Second, construct the $Q_\tau Q_\nu$ path responses corresponding to jth estimated path, denoted by $\{\hat{\mathbf{Y}}_j^{(q')}\}_{q'=0}^{Q_\tau Q_\nu - 1}$, and remove them from $\{\mathbf{Y}_{\text{pil}}^{(q')}\}_{q'=0}^{Q_\tau Q_\nu - 1}$. At the end of each iteration, we define a temporary matrix referred to as the residue matrix that contains the residue of $\{\mathbf{Y}_{\text{pil}}^{(q')}\}_{q'=0}^{Q_\tau Q_\nu - 1}$ after removal of the currently estimated path. The residue matrix is denoted by $\mathbf{W}_j^{(q')}$, where the subscript j is the iteration index and the superscript q' is the branch index, $q' \in \{0, 1, \ldots, Q_\tau Q_\nu - 1\}$. Let $\mathbf{w}_j^{(q')}$ denote the corresponding residue vector, obtained as $\mathbf{w}_j^{(q')}[kN + l] = \mathbf{W}_j^{(q')}[k, l]$. It is updated as $\mathbf{W}_{j+1}^{(q')} = \mathbf{W}_j^{(q')} - \hat{\mathbf{Y}}_j^{(q')}$ and used in the $(j+1)$th iteration to estimate the $(j+1)$th path. In the first iteration, i.e., $j = 1$, it is initialized with parallel branch matrices, i.e., $\mathbf{W}_1^{(q')} = \mathbf{Y}_{\text{pil}}^{(q')}, q' \in \{0, 1, \ldots, Q_\tau Q_\nu - 1\}$. The algorithm progresses until either the residue energy, defined as $E_{\text{res}} = \frac{\sum_{q'=0}^{Q_\tau Q_\nu - 1} ||\mathbf{W}_{j+1}^{(q')}||_2^2}{Q_\tau Q_\nu}$, where $||\cdot||_2$ denotes the Frobenius norm of matrix, is less than a threshold ϵ or the number of estimated paths exceeds $P_{\max}$, i.e., $j > P_{\max}$. The threshold ϵ is taken to be the average noise energy across the residue matrices of all the branches, i.e., $\epsilon = MNN_0$[5].

At the end of the algorithm, the estimated vectors of channel delays, Dopplers, and gains, i.e., $\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\nu}}$, and $\hat{\mathbf{h}}$, respectively, are used to construct the estimated effective channel matrices $\hat{\mathbf{H}}^{(q')}, q' \in \{0, 1, \ldots, Q_\tau Q_\nu - 1\}$, i.e., $(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\nu}}, \hat{\mathbf{h}})$ when used in (6) gives $\hat{h}(\tau, \nu)$, which when used in (11) gives $\hat{h}_{\text{dd}}(\tau, \nu)$,

[5]*Remark on the choice of ϵ:* Note that as the iterations progress by canceling the contribution of estimated paths, ideally, after all paths are estimated, the residue matrix will contain only noise. Hence, we have chosen the average noise energy across branches as the threshold and compute E_{res} as the average residual energy across branches post estimated path cancellation.

which when sampled and used in (26) gives $\hat{\mathbf{H}}^{(q')}, q' \in \{0,1,\ldots,Q_\tau Q_\nu-1\}$. The estimated $\hat{\mathbf{H}}^{(q')}$ matrices are input to the SSD algorithm in Sec. IV-A for detection. A detailed step-by-step description of the operations in the jth iteration of the channel estimation algorithm is provided below.

Step 1(a): *Estimation of delay and Doppler indices*

For the jth path, the algorithm finds the energy and location of the maximum energy bin among all the residue matrices in the jth iteration. This is done is two steps. First, the maximum energy bin and its location in each of the $\mathbf{W}_j^{(q')}$ matrix, $q' \in \{0,1,\ldots,Q_\tau Q_\nu-1\}$ are obtained as

$$\gamma_j^{(q')} = \max_{k\in\{0,\ldots,M-1\}, l\in\{0,\ldots,N-1\}} |\mathbf{W}_j^{(q')}[k,l]|^2, \quad (34)$$

$$(k_j^{(q')}, l_j^{(q')}) = \arg\max_{k\in\{0,\ldots,M-1\}, l\in\{0,\ldots,N-1\}} |\mathbf{W}_j^{(q')}[k,l]|^2, \quad (35)$$

respectively. Second, the branch whose matrix has the highest energy bin relative to the rest of the branches is obtained as

$$\tilde{q} = \arg\max_{q'\in\{0,\ldots,Q_\tau Q_\nu-1\}} [\gamma_j^{(q')}]. \quad (36)$$

The estimates of the delay and Doppler indices for the jth path, denoted as $\tilde{\mathbf{k}}[j]$ and $\tilde{\mathbf{l}}[j]$, respectively, are determined as

$$\tilde{\mathbf{k}}[j] = k_j^{(\tilde{q})} + \frac{\tilde{q}_\tau}{Q_\tau} - k_{\text{pil}}, \quad \tilde{\mathbf{l}}[j] = l_j^{(\tilde{q})} + \frac{\tilde{q}_\nu}{Q_\nu} - l_{\text{pil}}, \quad (37)$$

where $\tilde{q}_\tau = (\tilde{q})_{Q_\tau}$, $\tilde{q}_\nu = \lfloor \frac{\tilde{q}}{Q_\tau} \rfloor$, and $(\cdot)_K$ denotes modulo-K operation.

Step 1(b): *Refinement of delay/Doppler estimates*

A further refinement of the fractional estimates $\tilde{\mathbf{k}}[j]$ and $\tilde{\mathbf{l}}[j]$ obtained in **Step 1(a)** can be carried out using a low-complexity approach presented in Appendix B, by which a refined estimate of the jth path delay and Doppler indices, denoted by $\breve{\mathbf{k}}[j]$ and $\breve{\mathbf{l}}[j]$, is obtained as (see (58))

$$(\breve{\mathbf{k}}[j], \breve{\mathbf{l}}[j]) = \arg\max_{\mathcal{D}_k\times\mathcal{D}_l} \Big[{\mathbf{w}_j^{(\tilde{q})}}^H {\mathbf{R}^{(\tilde{q})}}^{-1} \mathbf{\Gamma}_j^{(\tilde{q})} ({\mathbf{\Gamma}_j^{(\tilde{q})}}^H {\mathbf{R}^{(\tilde{q})}}^{-1} \mathbf{\Gamma}_j^{(\tilde{q})})^{-1} {\mathbf{\Gamma}_j^{(\tilde{q})}}^H {\mathbf{R}^{(\tilde{q})}}^{-1} \mathbf{w}_j^{(\tilde{q})} \Big], \quad (38)$$

where $\mathcal{D}_k$ and $\mathcal{D}_l$ are the search spaces for delay and Doppler, $\mathbf{\Gamma}_j^{(\tilde{q})} = \mathbf{\Psi}_j^{(\tilde{q})}(d_k\Delta\tau, d_l\Delta\nu)\ \mathbf{x}_{\text{pil}}$, $d_k \in \mathcal{D}_k$, $d_l \in \mathcal{D}_l$, $\mathbf{\Psi}_j^{(\tilde{q})}(d_k\Delta\tau, d_l\Delta\nu)$ is obtained using (33), and $\mathbf{w}_j^{(\tilde{q})}[kN+l] = \mathbf{W}_j^{(\tilde{q})}[k,l]$. The search spaces $\mathcal{D}_k$ and $\mathcal{D}_l$ comprise of equally spaced points in the interval $\frac{1}{Q_\tau}$ and $\frac{1}{Q_\nu}$ around $\tilde{\mathbf{k}}[j]$ in delay axis and $\tilde{\mathbf{l}}[j]$ in Doppler axis, respectively, given by

$$\mathcal{D}_k = \left[\tilde{\mathbf{k}}[j] - \frac{1}{2Q_\tau} : \frac{1}{(|\mathcal{D}_k|-1)Q_\tau} : \tilde{\mathbf{k}}[j] + \frac{1}{2Q_\tau}\right] \quad (39)$$

along delay axis and

$$\mathcal{D}_l = \left[\tilde{\mathbf{l}}[j] - \frac{1}{2Q_\nu} : \frac{1}{(|\mathcal{D}_l|-1)Q_\nu} : \tilde{\mathbf{l}}[j] + \frac{1}{2Q_\nu}\right] \quad (40)$$

along the Doppler axis, where $|\mathcal{D}_k|$ and $|\mathcal{D}_l|$ denote the cardinality of the sets $\mathcal{D}_k$ and $\mathcal{D}_l$, respectively.

If the above refinement is performed, the estimates of delay and Doppler indices for jth path at the end of **Step 1** are $\hat{\mathbf{k}}[j] = \breve{\mathbf{k}}[j]$ and $\hat{\mathbf{l}}[j] = \breve{\mathbf{l}}[j]$, respectively. If the refinement is not performed, then $\hat{\mathbf{k}}[j] = \tilde{\mathbf{k}}[j]$ and $\hat{\mathbf{l}}[j] = \tilde{\mathbf{l}}[j]$. The estimated delays and Dopplers for the jth path are obtained as $\hat{\boldsymbol{\tau}}[j] = \hat{\mathbf{k}}[j]\Delta\tau$ and $\hat{\boldsymbol{\nu}}[j] = \hat{\mathbf{l}}[j]\Delta\nu$.

Step 2: *Estimation of channel gain*

Using the delay and Doppler estimates $\hat{\boldsymbol{\tau}}$ and $\hat{\boldsymbol{\nu}}$ from **Step 1**, the estimate of the channel gain is obtained using (59) in Appendix B as

$$\hat{\mathbf{h}}[j] = ({\hat{\mathbf{\Gamma}}_j^{(q')}}^H {\mathbf{R}^{(\tilde{q})}}^{-1} \hat{\mathbf{\Gamma}}_j^{(q')})^{-1} {\hat{\mathbf{\Gamma}}_j^{(q')}}^H {\mathbf{R}^{(\tilde{q})}}^{-1} \mathbf{w}_j^{(\tilde{q})}, \quad (41)$$

where $\hat{\mathbf{\Gamma}}_j^{(q')} = \hat{\mathbf{\Psi}}_j^{(\tilde{q})}(\hat{\boldsymbol{\tau}}[j], \hat{\boldsymbol{\nu}}[j])\ \mathbf{x}_{\text{pil}} \in \mathbb{C}^{MN\times 1}$.

Step 3: *Cancellation of estimated path*

Using $\hat{\mathbf{\Gamma}}_j^{(q')}$, the estimated response of q'th branch is obtained using (32) as $\hat{\mathbf{y}}_j^{(q')} = \hat{\mathbf{\Gamma}}_j^{(q')}\hat{\mathbf{h}}[j]$, and the corresponding matrix is obtained as $\hat{\mathbf{Y}}_j^{(q')}[k,l] = \hat{\mathbf{y}}_j^{(q')}[kN+l]$. The estimated path is removed and the residue matrix is updated as

$$\mathbf{W}_{j+1}^{(q')} = \mathbf{W}_j^{(q')} - \hat{\mathbf{Y}}_j^{(q')}. \quad (42)$$

The corresponding residue vector is updated as $\mathbf{w}_{j+1}^{(q')}[kN+l] = \mathbf{W}_{j+1}^{(q')}[k,l], \forall q' \in \{0,1,\ldots,Q_\tau Q_\nu-1\}$ and the residual energy is computed as $E_{\text{res}} = \frac{\sum_{q'=0}^{Q_\tau Q_\nu-1} ||\mathbf{W}_{j+1}^{(q')}||_2^2}{Q_\tau Q_\nu}$.

Step 4: *Stopping criteria*

The algorithm continues to estimate the paths iteratively by repeating **Steps 1-3** until either $j > P_{\max}$ or the residual energy goes below a threshold, i.e., $E_{\text{res}} \le \epsilon$, where $\epsilon = MNN_0$.

The channel estimation algorithm described above is listed in **Algorithm 2**. In Fig. 5, we present an iteration-wise illustration of the evolution of the residue matrices $\mathbf{W}_j^{(q')}$, $q' = 1,2,3,4$ for $j = 1,2,3$. A system with $M = N = 16$, pilot located at $k_{\text{pil}} = l_{\text{pil}} = 8$, pilot SNR of 20 dB, and $Q_\tau = Q_\nu = 2$ is considered. A two-path channel with unity gain ($P = 2$, $h_1 = h_2 = 1$), delays $\boldsymbol{\alpha} = [1.45, 0.65]$, and Dopplers $\boldsymbol{\beta} = [3.65, 1.95]$ is considered. It can be seen that, in the first iteration, significant inter-path interference is present in the residue matrices. However, as the algorithm proceeds to iteration 2, the interference profile of the residue matrices improves due to removal of the estimated first path. The improvement is even more significant in iteration 3 due to removal of the estimated second path as well. This leads to improved normalized mean square error (NMSE)[6] performance.

Remark on error propagation: The proposed channel estimation algorithm estimates channel paths sequentially in a path-by-path manner. While such iterative schemes may potentially suffer from error propagation as iterations progress, its impact is limited in the proposed approach. This can be observed in Fig. 5, where the energy in the residue matrix decreases as the iterations progress. This is because, 1) in each iteration, the path corresponding to the maximum energy bin across all over-sampled parallel branches is selected, ensuring that the strongest remaining path is estimated first, 2) DD-domain over-sampling improves energy localization for fractional DD paths, resulting in accurate estimation and effective cancellation, and 3) the use of a residual energy based stopping

[6]The NMSE across parallel branches is computed as $\frac{1}{Q_\tau Q_\nu}\sum_{q'=0}^{Q_\tau Q_\nu-1} \frac{||\hat{\mathbf{H}}^{(q')} - \mathbf{H}^{(q')}||_2^2}{||\mathbf{H}^{(q')}||_2^2}$.

Algorithm 2 Proposed channel estimation algorithm

Inputs: Received signal and noise covariance matrices on $Q_\tau Q_\nu$ branches : $\mathbf{Y}_{\text{pil}}^{(q')}$ and $\mathbf{R}^{(q')}$, pilot vector: $\mathbf{x}_{\text{pil}}$, maximum number of estimated paths $P_{\max}$, $\epsilon = MNN_0$
Initialize: path index $j=1$, residue matrix/vector in 1st iter: $\mathbf{W}_1^{(q')} = \mathbf{Y}_{\text{pil}}^{(q')}$, $\mathbf{w}_1^{(q')} = \mathbf{y}_{\text{pil}}^{(q')}$, $E_{\text{res}} = \frac{\sum_{q'=0}^{Q_\tau Q_\nu - 1} ||\mathbf{W}_1^{(q')}||_2^2}{Q_\tau Q_\nu}$
while $j < P_{\max}$ or $E_{\text{res}} \leq \epsilon$ **do**
  for $q' = 0$ to $Q_\tau Q_\nu - 1$ **do**
    $\gamma_j^{(q')} = \max\limits_{k\in\{0,\ldots,M-1\}, l\in\{0,\ldots,N-1\}} |\mathbf{W}_j^{(q')}[k,l]|^2$
    $[k_j^{(q')}, l_j^{(q')}] = \arg\max\limits_{k\in\{0,\ldots,M-1\}, l\in\{0,\cdots,N-1\}} |\mathbf{W}_j^{(q')}[k,l]|^2$
  end for
  Obtain $\tilde{q} = \arg\max\limits_{q'\in\{0,\ldots,Q_\tau Q_\nu - 1\}} [\gamma_j^{(q')}]$, $\tilde{q}_\tau = (\tilde{q})_{Q_\tau}$, $\tilde{q}_\nu = \lfloor \frac{\tilde{q}}{Q_\tau} \rfloor$
  $\tilde{\mathbf{k}}[j] = k_j^{(\tilde{q})} + \frac{\tilde{q}_\tau}{Q_\tau} - k_{\text{pil}}$, $\tilde{\mathbf{l}}[j] = l_j^{(\tilde{q}_\nu)} + \frac{\tilde{q}_\nu}{Q_\nu} - l_{\text{pil}}$
  Construct $\mathcal{D}_k = \left[\tilde{\mathbf{k}}[j] - \frac{1}{2Q_\tau} : \frac{1}{(|\mathcal{D}_k|-1)Q_\tau} : \tilde{\mathbf{k}}[j] + \frac{1}{2Q_\tau}\right]$ and $\mathcal{D}_l = \left[\tilde{\mathbf{l}}[j] - \frac{1}{2Q_\nu} : \frac{1}{(|\mathcal{D}_l|-1)Q_\nu} : \tilde{\mathbf{l}}[j] + \frac{1}{2Q_\nu}\right]$
  Obtain $(\breve{\mathbf{k}}[j], \breve{\mathbf{l}}[j])$ by solving (38), where $\mathbf{\Gamma}_j^{(\tilde{q})} = \mathbf{\Psi}_j^{(\tilde{q})}(d_k\Delta\tau, d_l\Delta\nu)\ \mathbf{x}_{\text{pil}}$, $\forall\ d_k \in \mathcal{D}_k$ and $d_l \in \mathcal{D}_l$.
  Obtain $(\hat{\mathbf{k}}[j], \hat{\mathbf{l}}[j]) = \begin{cases} (\tilde{\mathbf{k}}[j], \tilde{\mathbf{l}}[j]), \text{no refinement} \\ (\breve{\mathbf{k}}[j], \breve{\mathbf{l}}[j]), \text{with refinement} \end{cases}$
  Obtain $\hat{\boldsymbol{\tau}}[j] = \hat{\mathbf{k}}[j]\Delta\tau$ and $\hat{\boldsymbol{\nu}}[j] = \hat{\mathbf{l}}[j]\Delta\nu$
  Compute $\hat{\mathbf{h}}[j] = (\hat{\mathbf{\Gamma}}_j^{(q')^H} \mathbf{R}^{(\tilde{q})^{-1}} \hat{\mathbf{\Gamma}}_j^{(q')})^{-1} \hat{\mathbf{\Gamma}}_j^{(q')^H} \mathbf{R}^{(\tilde{q})^{-1}} \mathbf{w}_j^{(\tilde{q})}$, where $\hat{\mathbf{\Gamma}}_j^{(q')} = \hat{\mathbf{\Psi}}_j^{(\tilde{q})}(\hat{\boldsymbol{\tau}}[j], \hat{\boldsymbol{\nu}}[j])\ \mathbf{x}_{\text{pil}}$ and $\hat{\mathbf{\Psi}}_j^{(\tilde{q})}(\hat{\boldsymbol{\tau}}[j], \hat{\boldsymbol{\nu}}[j])$ is obtained using (33).
  Compute jth path contribution as $\hat{\mathbf{y}}_j^{(q')} = \hat{\mathbf{\Gamma}}_j^{(q')} \hat{\mathbf{h}}[j]$ and obtain $\hat{\mathbf{Y}}_j^{(q')}[k,l] = \hat{\mathbf{y}}_j^{(q')}[kN+l]$
  Update residue $\mathbf{W}_{j+1}^{(q')} = \mathbf{W}_j^{(q')} - \hat{\mathbf{Y}}_j^{(q')}$, $\mathbf{w}_{j+1}^{(q')}[kN+l] = \mathbf{W}_{j+1}^{(q')}[k,l]$, and $E_{\text{res}} = \frac{\sum_{q'=0}^{Q_\tau Q_\nu - 1} ||\mathbf{W}_{j+1}^{(q')}||_2^2}{Q_\tau Q_\nu}$
  update $j = j+1$
end
end while
Output: $\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\nu}}, \hat{\mathbf{h}}$

criterion prevents estimation of noise-dominated paths. This is evident from Fig. 6, where the NMSE performance of the proposed algorithm decreases as the number of iterations increases. Consequently, the BER performance with estimated CSI using the proposed scheme approaches the performance with perfect CSI (shown in the results in Sec. V).

V. Results and Discussions

In this section, we present the performance of the proposed detection and channel estimation algorithms. For this, we consider the following system parameters: $M = N = 16$, $\nu_p = 15$ kHz, $\tau_p = 1/\nu_p = 66.66$ μs, $B = M\nu_p = 240$ KHz, $T = N\tau_p = 1.06$ ms, and 4-QAM. We consider two channel models to evaluate the proposed algorithms under different channel conditions. The first channel model considers $P = 4$ paths and an exponential power delay profile (PDP). The maximum delay and Doppler spreads of the channel are taken to be $\tau_{\max} = 4\Delta\tau$ and $\nu_{\max} = 4\Delta\nu$, respectively, where $\Delta\tau = \frac{1}{B} = 4.16$ μs is the delay resolution and $\Delta\nu = \frac{1}{T} = 937.5$ Hz is the Doppler resolution. The path delays and Dopplers are such that $\tau_i \sim \text{Unif}[0, \tau_{\max}]$ and $\nu_i \sim \text{Unif}[-\nu_{\max}, \nu_{\max}]$, where $\text{Unif}[\cdot,\cdot]$ denotes uniform distribution. The channel gains follow exponential PDP given by $h_i \sim \mathcal{CN}(0, \sigma_i^2)$, where $\sigma_i^2 = \frac{e^{-0.1\frac{\tau_i}{\Delta\tau}}}{\sum_i e^{-0.1\frac{\tau_i}{\Delta\tau}}}$. The second channel model is the Vehicular A (Veh-A) channel model defined in the standard [23]. The PDP of the Veh-A channel model is given in Table I. The path Dopplers are distributed according to Jakes Doppler spectrum, i.e., $\nu_i = \nu_{\max}\cos(\theta)$, $\theta \in \text{Unif}[-\pi, \pi]$, and $\nu_{\max}$ is taken as 815 Hz. For channel estimation, the pilot symbol is placed at $(k_{\text{pil}}, l_{\text{pil}}) = (\frac{M}{2}, \frac{N}{2})$. The maximum number of paths estimated before the algorithm terminates is taken as $P_{\max} = 12$. The DD domain transmit filter $w_{\text{tx}}(\tau,\nu)$ is considered to be a sinc filter, given by

TABLE I: Veh-A channel PDP

Path index (i)	1	2	3	4	5	6
Delay τ_i (μs)	0	0.31	0.71	1.09	1.73	2.51
Path power (dB)	0	-1	-9	-10	-15	-20

$$w_{\text{tx}}(\tau,\nu) = \sqrt{BT}\text{sinc}(B\tau)\text{sinc}(T\nu). \tag{43}$$

The DD domain receive filter $w_{\text{rx}}(\tau,\nu)$ is matched to transmit filter, as shown in (9). The expression for the effective continuous DD channel response $h_{\text{dd}}(\tau,\nu)$ in (11) for the case of sinc filter is derived in Appendix C. Likewise, the covariance matrices of the filtered noise in (18) for DOS system and (24) for PBS system for sinc filter are derived in Appendix D. We consider $Q_\tau = Q_\nu = Q$ in all the simulations.

A. Performance/complexity of SSD algorithm

In Fig. 7, we present the BER performance of the proposed detection algorithm, assuming perfect knowledge of the I/O relation at the receiver. The channel model with exponential PDP is used. Results are shown for the following cases: i) conventional sampling ($Q = 1$) with MMSE detection, and ii) over-sampling ($Q = 2, 3, 4$) with SSD detection. We abbreviate the SSD algorithm as SSD-AB when the outputs of 'all branches' are used and as SSD-SB when the 'subset of branches' (as described in Sec. IV-A1) is used. In addition to the performance of SSD-AB and SSD-SB, the performance of the joint MMSE detection (abbreviated as JD) carried out on the full DOS vector is provided. From Fig. 7, it is seen that the performance obtained with over-sampling using the proposed SSD algorithm and JD are superior to that obtained by performing MMSE detection with conventional sampling. With $Q = 2$, SNR gains of about 4 dB, 4 dB, and 2.5 dB at 10^{-3} BER are achieved by performing JD, SSD-AB, and SSD-SB, respectively, when compared to MMSE detection using conventional sampling. Note that although the performance of JD and SSD-AB are comparable for $Q = 2$, the performance with JD saturates for $Q > 2$, while that of SSD-AB improves with increasing sampling factors of $Q = 3, 4$. This is due to the high covariance in the full DOS noise vector (used for JD) compared to that with PBS (used for SSD-AB and SSD-SB). Further, the performance of SSD-AB is superior compared to that obtained with SSD-SB. This is because, while all branches

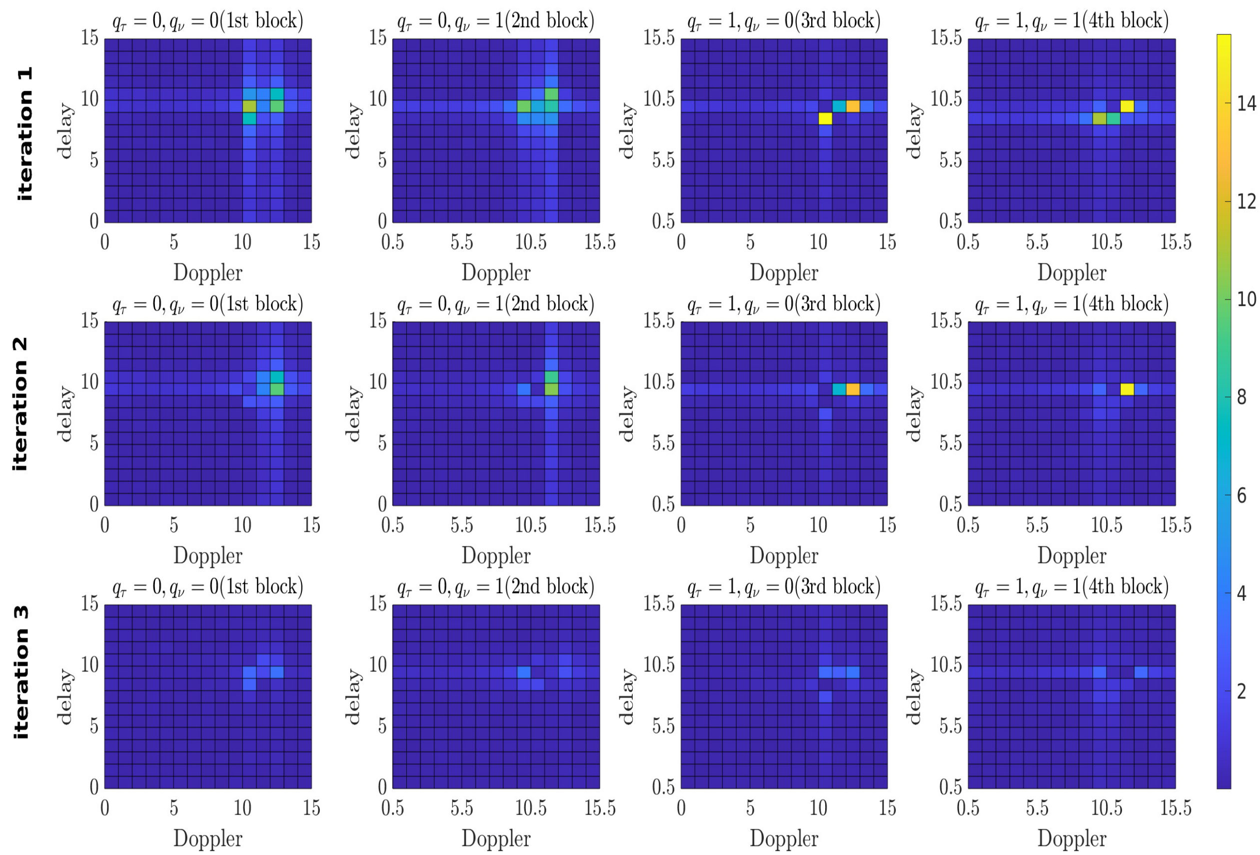


Fig. 5: Illustration of the evolution of the channel estimation algorithm as the iterations progress for $M = N = 16$, $P = 2$, $\mathbf{h} = [1, 1]$, $\boldsymbol{\alpha} = [1.45, 0.65]$, $\boldsymbol{\beta} = [3.65, 1.95]$, $k_{\text{pil}} = l_{\text{pil}} = 8$, and $Q_\tau = Q_\nu = 2$.

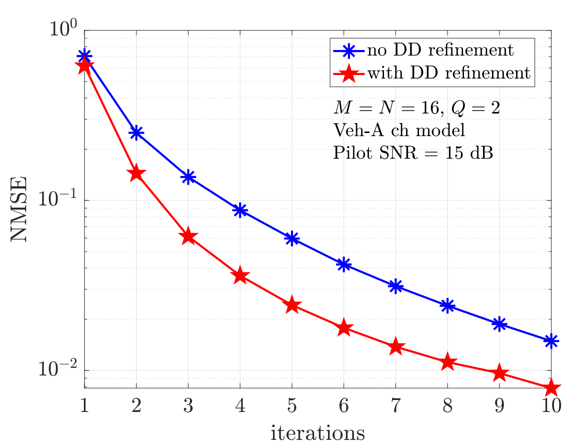


Fig. 6: NMSE performance of proposed estimation algorithm as a function of number of iterations.

are processed in SSD-AB, only a select subset of branches is processed in SSD-SB.

Complexity: Recall from Sec. IV-A and Sec. IV-A1 that JD, SSD-AB, and SSD-SB algorithms have complexities of $\mathcal{O}(Q^6M^3N^3)$, $Q^2\mathcal{O}(M^3N^3)$, and $Q\mathcal{O}(M^3N^3)$, respectively. That is, the complexity grows exponentially, quadratically, and linearly in Q for JD, SSD-AB, and SSD-SB algorithms, respectively. The simulation results in Fig. 7 show that SSD-SB with $Q = 4$ (4 parallel branches) achieves performance which is very close to those of SSD-AB algorithm with $Q = 4$ (16 parallel branches) and JD algorithm, making the proposed SSD-SB algorithm attractive from both performance and complexity perspectives.

B. Performance/complexity of channel estimation algorithm

The NMSE and the corresponding BER performance of the proposed channel estimation algorithm (detailed in **Algorithm 2**) for $Q = 1$ are shown in Figs. 8a and 8b, respectively. We abbreviate the proposed algorithm as 'with refinement (WR)' if the refinement step (Step 1 (b) in **Algorithm 2**) is included and as 'no refinement (NR)' if Step 1 (b) is not included. The Veh-A channel model is used. For comparison purposes, the performance of the model-free (MFr) and least-squares-based model-dependent (MD-LS) channel estimation algorithms reported in [7] are also included. MMSE detection is used to obtain the BER performance. The NMSE and BER performance of the proposed algorithm with no refinement (NR) is plotted as a function of pilot SNR in Fig. 8a and 8b.

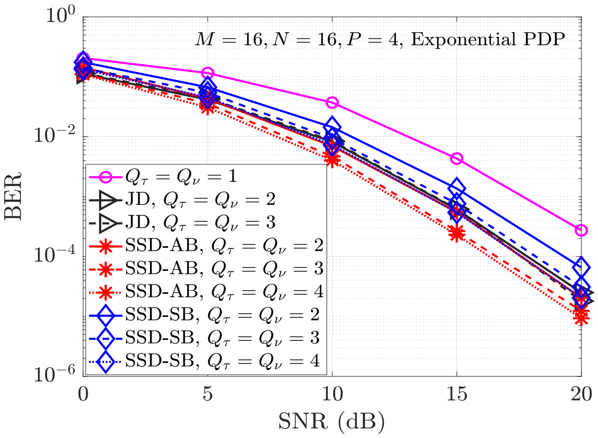
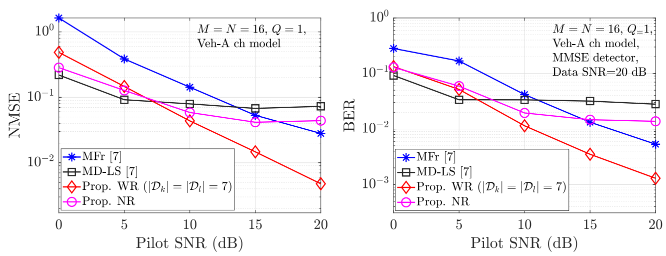


Fig. 7: BER performance of the proposed SSD algorithm (SSD-AB, SSD-SB) with perfect channel knowledge.

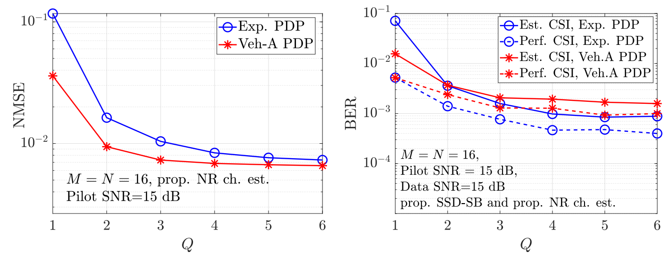


(a) NMSE vs pilot SNR (b) BER vs pilot SNR

Fig. 8: NMSE and BER performance of the proposed channel estimation algorithm for $Q = 1$ in comparison with MFr and MD-LS algorithms in [7].

It is seen that the proposed NR algorithm has a superior NMSE and BER performance when compared to MFr algorithm up to a pilot SNR of 15 dB, beyond which the performance of MFr algorithm is good. In Fig. 8a, the NMSE performance of the proposed algorithm with refinement (WR) is also depicted for $|\mathcal{D}_k| = |\mathcal{D}_l| = 7$ as a function of pilot SNR. The proposed WR algorithm is found to achieve better NMSE performance compared to the MFr and MD-LS algorithms. This is attributed to its ability to estimate fractional delays and Dopplers accurately, whereas the MFr algorithm is noise-limited and the MD-LS algorithm is constrained by imprecise integer estimates of path delays and Dopplers. The better NMSE performance of the proposed WR algorithm translates into better BER performance, as shown in Fig. 8b.

Complexity: Table II presents the worst-case complexity of the proposed algorithm in terms of the number of real multiplications and additions involved. For comparison, the complexities of the MFr and MD-LS algorithms are also included. For the specific parameter values of $M = N = 16$, $P = 6$, $P_{\max} = 12$, $|\mathcal{D}_k| = |\mathcal{D}_l| = 7$, $Q = 1$, the resultant complexity for MFr, MD-LS, proposed WR, and proposed NR algorithms are $8.192 \times 10^6, 4.522 \times 10^8, 3.5275 \times 10^9$, and



(a) NMSE vs Q (b) BER vs SNR

Fig. 9: NMSE and BER performance of the proposed receiver as a function of the over-sampling factor Q.

1.0712×10^9, respectively[7]. Considering the case of PBS with $Q = 2$, the complexities of the proposed WR and proposed NR algorithms increase to 4.9184×10^9 and 2.7416×10^9, respectively. The proposed NR algorithm with $Q = 2$ has a lower complexity than the proposed WR algorithm with $Q = 1$. Furthermore, the performance results reported next (see Fig. 11) show that the proposed NR algorithm with $Q = 2$ over-sampling and a computational complexity of 2.7416×10^9 performs better by about 1.5 dB at a BER of 1.2×10^{-3} compared to the proposed WR algorithm with $Q = 1$ and a complexity of 3.5275×10^9, highlighting the effectiveness of the NR algorithm with $Q > 1$ as a good low-complexity approach achieving good performance.

C. NMSE/BER performance as a function of Q and SNR

In Figs. 9a and 9b, we present the NMSE and BER performance of the proposed low-complexity receiver incorporating both the SSD-SB detection algorithm and the NR algorithm for channel estimation. Results for exponential PDP channel model and Veh-A channel model are presented. In Fig. 9a, the NMSE performance of the NR algorithm is shown for a pilot SNR of 15 dB as a function of over-sampling factor Q. It is seen that the NMSE performance of the NR algorithm improves as the over-sampling factor increases. This is attributed to the increasingly more accurate fractional estimates of path delays and Dopplers for increasing values of Q, which helps to obtain better estimates of channel gains, resulting in an improved estimate of $\mathbf{H}$. The improvement in performance beyond $Q = 3$ for Veh-A channel model and $Q = 4$ for exponential channel model is marginal. In Fig. 9b, the BER performance of the NR algorithm is shown in comparison with that obtained using perfect channel knowledge. A data SNR of 15 dB is considered for illustration. It can be observed that the BER performance with both estimated channel state information (CSI) and perfect CSI improve with Q and the performance with estimated CSI approaches that obtained with perfect CSI as Q increases. Further, the performance gain is only marginal as Q increases beyond 3 and 4 for Veh-A and exponential channel models, respectively.

[7]When constructing the $\mathbf{H}$ matrix using (26), the effective channel is constrained to span two DD periods on either side of the fundamental period along both the delay and Doppler axis, such that $|m| \leq 2$ and $|n| \leq 2$.

TABLE II: Complexity of the channel estimation algorithms.

Algorithm	Number of real multiplications	Number of real additions	$M = N = 16$, $P = 6$, $Q = 1$ $\lvert\mathcal{D}_k\rvert = \lvert\mathcal{D}_l\rvert = 7$	$M = N = 16$, $P = 6$, $Q = 2$ $\lvert\mathcal{D}_k\rvert = \lvert\mathcal{D}_l\rvert = 7$
MFr	$100M^2N^2$	$25M^2N^2$	8.192×10^6	-
MD-LS	$50M^2N^2(15P+6)+$ $MN(2P^2+P)+P^3+P$	$350M^2N^2P+MN$ $(P^2+2P-2)+P^3-3P$	4.522×10^8	-
Prop. WR	$M^3N^3+P_{\max}[M^2N^2$ $(41d^2+529Q_\tau Q_\nu+16)+$ $12MN(d^2+1)+5Q_\tau Q_\nu+12]+$ $25M^2N^2(15P_{\max}+16)$	$M^3N^3+P_{\max}$ $[M^2N^2(16d^2+175Q_\tau Q_\nu)+$ $4MNd^2+Q_\tau Q_\nu-4d^2+2]+$ $175M^2N^2P_{\max}$	3.5275×10^9	4.9184×10^9
Prop. NR	$M^3N^3+P_{\max}$ $[M^2N^2(529Q_\tau Q_\nu+16)+$ $12MN+5Q_\tau Q_\nu+8]+$ $25M^2N^2(15P_{\max}+16)$	$M^3N^3+P_{\max}[M^2N^2(179$ $Q_\tau Q_\nu+12)+12MN+$ $Q_\tau Q_\nu]+175M^2N^2P_{\max}$	1.0712×10^9	2.7416×10^9

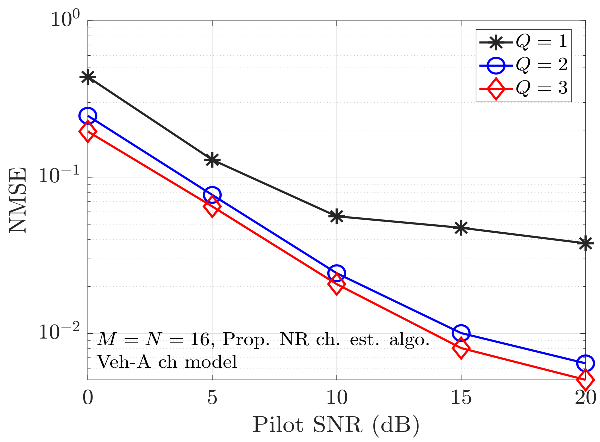


Fig. 10: NMSE performance of the proposed NR channel estimation algorithm for different over-sampling factors.

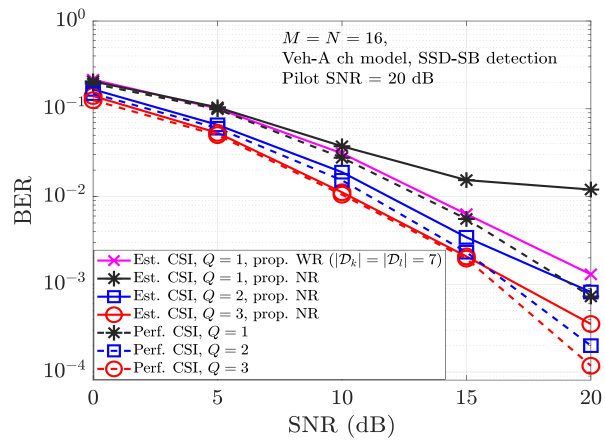


Fig. 11: BER performance of the proposed receiver for different over-sampling factors.

In Figs. 10 and 11, we show the NMSE and BER performance of the proposed low-complexity receiver as a function of SNR for Veh-A channel model. In Fig. 10, the NMSE performance is shown as a function of pilot SNR for $Q = 1, 2, 3$. It is observed that the NMSE performance improves as pilot SNR increases. In addition, the performance improves substantially with over-sampling factor. In Fig. 11, the BER performance of proposed algorithm is shown as a function of data SNR for a pilot SNR of 20 dB. It is observed that as Q increases, the performance with estimated CSI gets closer to the corresponding performance obtained with perfect CSI.

D. Diversity performance

For diversity analysis, the I/O relation in (20) is alternately written in the form $\mathbf{y}_{\text{dos}} = \mathbf{X}\mathbf{h} + \mathbf{n}_{\mathbf{dos}}$, where $\mathbf{X} = [\mathbf{\Psi}_1(\tau_1,\nu_1) \ \cdots \ \mathbf{\Psi}_P(\tau_p,\nu_p)](\mathbf{I}_P \otimes \mathbf{x}) \in \mathbb{C}^{MNQ_\tau Q_\nu \times P}$ and $\mathbf{\Psi}_i(\tau_i,\nu_i) = \mathbf{H}_{\text{dos}}$ with $h_i = 1, \tau_u = \nu_u = h_u = 0, \forall\ u \neq i,\ u \in \{1,\cdots,P\}$, $\mathbf{I}_P$ denotes $P \times P$ identity matrix, and $\otimes$ denotes Kronecker product. Note that $\mathbf{X}$ is a function of transmitted vector, path delays and Dopplers, and $\mathbf{h} = [h_1 h_2 \cdots h_P]^T$ is a $P \times 1$ vector. Then, the diversity order (DO) achieved by the system in (20) is given by the minimum rank of the difference matrices [21], i.e.,

$$\text{DO} = \min_{i,j \quad i \neq j} \ \text{rank}(\mathbf{X}_i - \mathbf{X}_j), \tag{44}$$

where $\mathbf{X}_i$ corresponds to the transmitted vector $\mathbf{x}_i \in \mathbb{A}^{MN}, i \in \{1, 2, \cdots, |\mathbb{A}^{MN}|\}$. Table III shows the rank profile of the difference matrices $\mathbf{X}_i - \mathbf{X}_j$ for different DD channel profiles (A, B, C, D) with conventional sampling and over-sampling. It is observed that the minimum rank obtained in all cases is equal to the number of paths, i.e., P. Hence, the full DD diversity of P is achieved with conventional sampling and over-sampling. This result is further corroborated through BER simulations in Fig. 12 for DD profile-B. The diversity slopes of the BER versus SNR curves for conventional sampling and over-sampling are found to be 2 (the same as the number of DD paths), which agrees with the diversity order observed through the rank profile in Table III.

VI. Conclusions

We considered a DD domain over-sampling framework for Zak-OTFS receivers. We presented an analysis using a minimum distance measure and PEP that indicated improved

TABLE III: Rank profile of the difference matrices.

System parameters			Number of difference matrices with rank 1	2	3	4	Minimum rank
$M=2, N=2, P=2$ $\tau_{\max}=1.2\Delta\tau, \nu_{\max}=1.2\Delta\nu$ $Q_\tau=Q_\nu=1,2$	DD profile-A	$\tau=[0,1]\Delta\tau$ $\nu=[0,1]\Delta\nu$	0	240	-	-	2
	DD profile-B	$\tau=[0.2,1.1]\Delta\tau$ $\nu=[-0.4,0.6]\Delta\nu$	0	240	-	-	2
$M=4, N=2, P=4$ $\tau_{\max}=3\Delta\tau, \nu_{\max}=1.2\Delta\nu$ $Q_\tau=Q_\nu=1,2$	DD profile-C	$\tau=[0,1,2,3]\Delta\tau$ $\nu=[0,1,0,1]\Delta\nu$	0	0	0	65280	4
	DD profile-D	$\tau=[0.2,1.1,2.8,1.6]\Delta\tau$ $\nu=[-0.3,1.2,0.8,-0.2]\Delta\nu$	0	0	0	65280	4

performance with over-sampling compared to conventional sampling. We proposed detection and channel estimation algorithms to reap this performance improvement in a low complexity manner. To achieve low-complexity detection in the over-sampling framework, a parallel branch sampling, which can be viewed as conventional sampling of delay- and Doppler-shifted signals, was proposed. A novel partitioning of the frame along with processing of a subset of partitions was shown to achieve improved detection performance with over-sampling. Applying over-sampling and partitioning on the received pilot frame, we devised a low-complexity algorithm to estimate the delays, Dopplers, and gains of the DD channel taps based on maximum energy across partitions. The proposed algorithm was shown to achieve better NMSE performance compared to the model-free approach of I/O relation estimation. The proposed channel estimation algorithm, along with the proposed detection algorithm, was shown to achieve good BER performance.

APPENDIX A
MOTIVATION/RATIONALE FOR OVER-SAMPLING

In this Appendix, we present an analysis that compares the performance of conventional sampling and over-sampling in Zak-OTFS using a minimum distance measure and PEP.

A. Analysis

Dropping the subscripts in (20) for notational simplicity, we write the DD domain received vector for conventional sampling, i.e., when $Q_\tau = Q_\nu = 1$, as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, (45) \tag{45}$$

where $\mathbf{y}$ and $\mathbf{x}$ are the received and transmitted vectors, respectively, $\mathbf{H}$ is the effective channel matrix, and $\mathbf{n} \sim \mathcal{CN}(\mathbf{0}, \mathbf{R}_{nn})$ is the additive noise. The ML detection rule for estimating the transmitted vector is

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \ (\mathbf{y} - \mathbf{H}\mathbf{x})^H \mathbf{R}_{nn}^{-1} (\mathbf{y} - \mathbf{H}\mathbf{x}). \tag{46}$$

Let $\mathbf{x}_i, i \in 1, 2, \cdots, |\mathbb{A}|^{MN}$ denote the set of all possible OTFS transmit vectors of length MN. Defining $\mathbf{a} = \mathbf{H}(\mathbf{x}_i - \mathbf{x}_j)$ and $d_{ij} = \mathbf{a}^H \mathbf{R}_{nn}^{-1} \mathbf{a}$, the PEP that the receiver will decide in favor of $\mathbf{x}_j$ when $\mathbf{x}_i$ is transmitted is [21]

$$P(\mathbf{x}_i \to \mathbf{x}_j | \mathbf{x}_i \text{ sent}, \mathbf{H}) = Q\left(\sqrt{d_{ij}/2}\right). \tag{47}$$

Note that as d_{ij} increases, the PEP decreases (due to monotonicity of the Q-function). Now, consider the case when over-sampling is performed on the received signal in delay and/or Doppler domain, i.e., $Q_\tau \geq 1$ and/or $Q_\nu \geq 1$. Let $\mathbf{y}'$, $\mathbf{n}'$, and $\mathbf{H}'$ denote the vectors and matrix consisting of the additional samples of received signal, noise, and channel, respectively, resulting due to over-sampling, i.e., samples at points in between the information grid points. The resultant overall over-sampled signal can be written as

$$\mathbf{y}_{\text{os}} = \begin{bmatrix} \mathbf{y} \\ \mathbf{y}' \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{H}' \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{n} \\ \mathbf{n}' \end{bmatrix} = \mathbf{H}_{\text{os}}\mathbf{x} + \mathbf{n}_{\text{os}}. \tag{48}$$

The covariance matrix of $\mathbf{n}_{\text{os}}$ is given by $\mathbf{R}_{\text{os}} = \mathbb{E}[\mathbf{n}_{\text{os}}\mathbf{n}_{\text{os}}^H] = \begin{bmatrix} \mathbf{R}_{nn} & \mathbf{R}_{nn'} \\ \mathbf{R}_{nn'}^H & \mathbf{R}_{n'n'} \end{bmatrix}$. The inverse of $\mathbf{R}_{\text{os}}$ can be obtained using block matrix inversion formula as [22]

$$\mathbf{R}_{\text{os}}^{-1} = \begin{bmatrix} \mathbf{R}_{nn}^{-1} + \mathbf{R}_{nn}^{-1}\mathbf{R}_{nn'}\mathbf{S}^{-1}\mathbf{R}_{nn'}^H\mathbf{R}_{nn}^{-1} & -\mathbf{R}_{nn}^{-1}\mathbf{R}_{nn'}\mathbf{S}^{-1} \\ -\mathbf{S}^{-1}\mathbf{R}_{nn'}^H\mathbf{R}_{nn}^{-1} & \mathbf{S}^{-1} \end{bmatrix},$$

where $\mathbf{S} = \mathbf{R}_{n'n'} - \mathbf{R}_{nn'}^H\mathbf{R}_{nn}^{-1}\mathbf{R}_{nn'}$ is the Schur complement of $\mathbf{R}_{nn}$. The ML estimate of the transmitted vector and the PEP for the over-sampled case are obtained as

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} (\mathbf{y}_{\text{os}} - \mathbf{H}_{\text{os}}\mathbf{x})^H \mathbf{R}_{\text{os}}^{-1} (\mathbf{y}_{\text{os}} - \mathbf{H}_{\text{os}}\mathbf{x}), \tag{49}$$

$$P(\mathbf{x}_i \to \mathbf{x}_j | \mathbf{x}_i \text{ sent}, \mathbf{H}_{\text{os}}) = Q\left(\sqrt{d'_{ij}/2}\right), \tag{50}$$

where $d'_{ij} = (\mathbf{H}_{\text{os}}(\mathbf{x}_i - \mathbf{x}_j))^H \mathbf{R}_{\text{os}}^{-1} (\mathbf{H}_{\text{os}}(\mathbf{x}_i - \mathbf{x}_j))$. With the above, we make and prove the following claim.

Claim: PEP is lower in case of over-sampled receiver when compared to conventionally sampled receiver.

Proof: Let $\mathbf{a}' = \mathbf{H}'(\mathbf{x}_i - \mathbf{x}_j)$. Then, d'_{ij} can be simplified as

$$\begin{aligned} d'_{ij} = \ & \mathbf{a}^H\mathbf{R}_{nn}^{-1}\mathbf{a} + \mathbf{a}^H\mathbf{R}_{nn}^{-1}\mathbf{R}_{nn'}\mathbf{S}^{-1}\mathbf{R}_{nn'}^H\mathbf{R}_{nn}^{-1}\mathbf{a} - \\ & \mathbf{a}^H\mathbf{R}_{nn}^{-1}\mathbf{R}_{nn'}\mathbf{S}^{-1}\mathbf{a}' - \\ & \mathbf{a}'^H\mathbf{S}^{-1}\mathbf{R}_{nn'}^H\mathbf{R}_{nn}^{-1}\mathbf{a} + \mathbf{a}'^H\mathbf{S}^{-1}\mathbf{a}' \\ = \ & d_{ij} + (\mathbf{R}_{nn'}^H\mathbf{R}_{nn}^{-1}\mathbf{a} - \mathbf{a}')^H \mathbf{S}^{-1} (\mathbf{R}_{nn'}^H\mathbf{R}_{nn}^{-1}\mathbf{a} - \mathbf{a}'). \end{aligned} \tag{51}$$

Note that the quadratic term in (51) is non-negative since $\mathbf{S}$ is positive semi-definite. Hence, $d'_{ij} \geq d_{ij}$ and $Q\big(\sqrt{d'_{ij}/2}\big) \leq Q\big(\sqrt{d_{ij}/2}\big)$. That is, the over-sampled receiver can have lower PEP and hence lower average probability of error compared to the conventionally sampled receiver. We verify this through computation of PEP-based analytical BER upper bound (UB) obtained using union bound and averaging over fading distribution as well as BER simulation in Fig. 12, as described below.

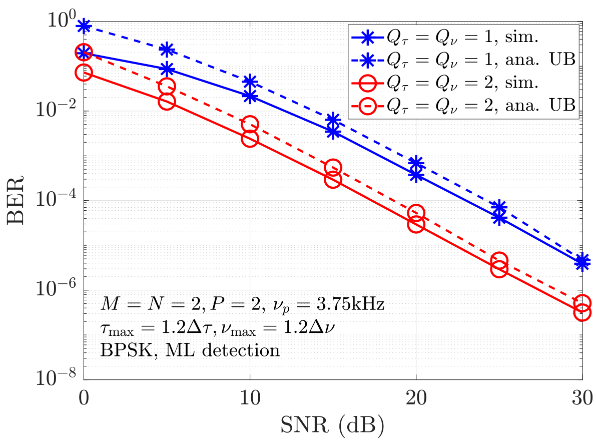


Fig. 12: BER performance comparison between conventional sampling and over-sampling.

B. BER performance

In Fig. 12, we show the BER performance of Zak-OTFS for conventional sampling ($Q_\tau = Q_\nu = 1$) and direct over-sampling ($Q_\tau = Q_\nu = 2$) under ML detection with perfect CSI. BER upper bound (UB) obtained using PEPs as well as simulated BER are plotted. The following system parameters are used: $M = N = 2, P = 2, \nu_p = 3.75$ kHz, $\tau_p = \frac{1}{\nu_p} = 0.2667$ ms, $\tau_{\max} = 1.2\Delta\tau$ and $\nu_{\max} = 1.2\Delta\nu$. The DD pairs, i.e, (τ_i, ν_i)s, of the two paths are fixed as $(0.2\Delta\tau, -0.4\Delta\nu)$ and $(1.1\Delta\tau, 0.6\Delta\nu)$, and the channel gains are distributed as $h_i \sim \mathcal{CN}(0, \frac{1}{P})$. The symbols are drawn from BPSK constellation. From Fig. 12, it is observed that over-sampling achieves better BER performance compared to conventional sampling, which is consistent with the analytical prediction in the previous subsection.

Appendix B
Refinement of delay-Doppler estimates of jth path

In this Appendix, we present a low-complexity framework that enables refinement of the estimates of the delay and Doppler indices obtained in **Step 1(a)** of the channel estimation algorithm. For notational simplicity, we drop the frame type indication denoted by subscript 'pil' in input-output relation in (32). Then, for the q'th branch, we have

$$\mathbf{y}^{(q')} = \mathbf{\Phi}^{(q')}\mathbf{h} + \mathbf{n}^{(q')}, \tag{52}$$

where $\mathbf{n}^{(q')} \sim \mathcal{CN}(\mathbf{0}, \mathbf{R}^{(q')})$ and $\mathbf{R}^{(q')}$ is the noise covariance matrix corresponding to the q'th branch. The conditional distribution $p_{\text{Y}}(\mathbf{y}^{(q')}|\boldsymbol{\tau}, \boldsymbol{\nu}, \mathbf{h}) \sim \mathcal{CN}(\mathbf{\Phi}\mathbf{h}, \mathbf{R}^{(q')})$ is given by

$$p_{\text{Y}}(\mathbf{y}^{(q')}|\boldsymbol{\tau}, \boldsymbol{\nu}, \mathbf{h}) = \frac{1}{\pi^{MN}\det(\mathbf{R}^{(q')})} e^{-(\mathbf{y}^{(q')}-\mathbf{\Phi}^{(q')}\mathbf{h})^H \mathbf{R}^{(q')^{-1}}(\mathbf{y}^{(q')}-\mathbf{\Phi}^{(q')}\mathbf{h})}, \tag{53}$$

where $\det(\cdot)$ denotes determinant operator, and the ML estimate of $(\boldsymbol{\tau}, \boldsymbol{\nu}, \mathbf{h})$ is given by

$$(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\nu}}, \hat{\mathbf{h}}) = \arg \max_{(\boldsymbol{\tau},\boldsymbol{\nu},\mathbf{h})} p_{\text{Y}}(\mathbf{y}^{(q')}|\boldsymbol{\tau}, \boldsymbol{\nu}, \mathbf{h}). \tag{54}$$

The optimization problem in (54) can be solved to obtain [20]

$$(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\nu}}) = \arg \max_{(\boldsymbol{\tau},\boldsymbol{\nu})} \big[\mathbf{y}^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Phi}^{(q')} (\mathbf{\Phi}^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Phi}^{(q')})^{-1} \mathbf{\Phi}^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{y}^{(q')}\big], \tag{55}$$

$$\hat{\mathbf{h}} = (\mathbf{\Phi}^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Phi}^{(q')})^{-1} \mathbf{\Phi}^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{y}^{(q')}, \tag{56}$$

where $(\boldsymbol{\tau}, \boldsymbol{\nu})$ is the joint DD search space. Note that $\hat{\boldsymbol{\tau}}$ and $\hat{\boldsymbol{\nu}}$ are P-dimensional vectors. Hence, obtaining joint estimates $(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\nu}})$ by solving optimization problem in (55) is computationally complex. The term $(\mathbf{\Phi}^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Phi}^{(q')})^{-1}$ in (55) is diagonally dominant. Hence, approximating the off-diagonal terms in $(\mathbf{\Phi}^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Phi}^{(q')})^{-1}$ to zero [20], (55) can be written as

$$(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\nu}}) = \arg \max_{(\boldsymbol{\tau},\boldsymbol{\nu})} \sum_{i=1}^{P} \mathbf{y}^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Gamma}_i^{(q')} (\mathbf{\Gamma}_i^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Gamma}_i^{(q')})^{-1} \mathbf{\Gamma}_i^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{y}^{(q')}, \tag{57}$$

where $\mathbf{\Gamma}_i^{(q')} = \mathbf{\Psi}^{(q')}(\tau_i, \nu_i)\mathbf{x}_{\text{pil}}$ (see (33)). In (57), observe that the objective function depends on ith path parameters (τ_i, ν_i) only through ith term in the summation. Hence, the estimation of delay, Doppler, and channel gains of the different channel paths can be carried out separately. Thus, we obtain an estimate of ith path parameters by using path separability in (55) and (56) as

$$(\hat{\boldsymbol{\tau}}[j], \hat{\boldsymbol{\nu}}[j]) = \arg \big[\max_{\tau\in\mathbb{R}^+, \nu\in\mathbb{R}} \mathbf{w}_i^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Gamma}_i^{(q')} (\mathbf{\Gamma}_i^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Gamma}_i^{(q')})^{-1} \mathbf{\Gamma}_i^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{w}_i^{(q')}\big], \tag{58}$$

and

$$\hat{\mathbf{h}}[j] = (\mathbf{\Gamma}_j^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{\Gamma}_j^{(q')})^{-1} \mathbf{\Gamma}_j^{(q')^H} \mathbf{R}^{(q')^{-1}} \mathbf{w}_j^{(q')}, \tag{59}$$

where $\mathbf{\Gamma}_j^{(q')} = \mathbf{\Psi}_j^{(q')}(\tau, \nu)\ \mathbf{x}_{\text{pil}}$ and $\mathbf{w}_j^{(q')}$ is the residue vector in jth iteration.

Appendix C
Effective channel response in DD domain

The expression for the effective channel response (see (11)) when sinc filter is used is derived here. Considering $a(\tau, \nu) = h(\tau, \nu) *_\sigma w_{\text{tx}}(\tau, \nu)$ and using $w_{\text{tx}}(\tau, \nu)$ given by (43) and the definition of twisted convolution gives

$$\begin{aligned} a(\tau, \nu) &= h(\tau, \nu) *_\sigma w_{\text{tx}}(\tau, \nu) \\ &= \sqrt{BT} \sum_{i=1}^{P} h_i e^{-j2\pi\tau_i\nu_i} \text{sinc}(B(\tau - \tau_i)) \\ &\quad \text{sinc}(T(\nu - \nu_i)) e^{j2\pi\tau\nu_i}. \end{aligned} \tag{60}$$

Using (60) in (11) and simplifying, we get

$$\begin{aligned} h_{\text{dd}}(\tau, \nu) &= w_{\text{rx}}(\tau, \nu) *_\sigma a(\tau, \nu) \\ &= BT \sum_{i=1}^{P} h_i e^{-j2\pi\tau_i\nu_i} I_1 I_2, \end{aligned} \tag{61}$$

where

$$I_1 = \int_{\tau'} \text{sinc}(B\tau')\text{sinc}(B(\tau - \tau' - \tau_i)) e^{j2\pi(\tau-\tau')\nu_i} d\tau', \tag{62}$$

$$A(\tau_1,\nu_1,\tau_2,\nu_2) = \frac{B\tau_p}{T}\sum_{q_1=-\infty}^{\infty}\sum_{q_2=-\infty}^{\infty} e^{-j2\pi q_1\tau_p\nu_1}\text{rect}(\frac{\tau_1+q_1\tau_p}{T})e^{-j2\pi q_2\tau_p\nu_2}\text{rect}(\frac{\tau_2+q_2\tau_p}{T}) \\ \int_{\tau_1'}\int_{\tau_2'}\text{sinc}(B\tau_1')\text{sinc}(B\tau_2')\mathbb{E}[n(\tau_1-\tau_1'+q_1\tau_p)n^*(\tau_2-\tau_2'+q_2\tau_p)]d\tau_2'd\tau_1'. \quad (67)$$

and

$$I_2 = \int_{\nu'}\text{sinc}(T\nu')\text{sinc}(T(\nu-\nu'-\nu_i))e^{j2\pi\tau\nu'}d\nu'. \quad (63)$$

It is noted that a general expression for $h_{\text{dd}(\tau,\nu)}$ for an arbitrary pulse-shaping filter in the matched filtering configuration is presented in [12] (see Eq. (57) in [12]). This $h_{\text{dd}(\tau,\nu)}$ expression in [12] is in a form of two separable integrals, corresponding to the auto-ambiguity functions of the time and frequency domain representations of the Doppler and delay domain components of the pulse-shaping filter, respectively. Observe that the integrals in (61) are similar to those of the auto-ambiguity integrals in [12]. Here, for the case of sinc filter, we further simplify (62) and (63) using the Plancheral's theorem and substitute them in (61) to obtain the closed-form expression as

$$h_{\text{dd}}(\tau,\nu) = \frac{1}{BT}\sum_{i=1}^{P}h_i e^{j\pi(\tau-\tau_i)\nu_i}e^{j\pi(\nu-\nu_i)\tau} \\ (T-|\tau|)(B-|\nu_i|)\text{sinc}\left((\nu-\nu_i)(T-|\tau|)\right) \\ \text{sinc}\left((\tau-\tau_i)(B-|\nu_i|)\right)\mathbb{1}_{\{|\tau|<T\}}\mathbb{1}_{\{|\nu_i|<B\}}. \quad (64)$$

Appendix D
Covariance of Noise Samples

The covariance of the filtered noise samples in the DD domain for sinc filter is derived here. The filtered noise is given by $n_{\text{dd}}^{w_{\text{rx}}}(\tau,\nu) = w_{\text{rx}}(\tau,\nu) *_\sigma n_{\text{dd}}(\tau,\nu)$. Using $w_{\text{rx}}(\tau,\nu) = w_{\text{tx}}(-\tau,-\nu)e^{j2\pi\tau\nu}$ in $n_{\text{dd}}^{w_{\text{rx}}}(\tau,\nu)$, where $w_{\text{tx}}(\tau,\nu)$ is given by (43), we obtain

$$n_{\text{dd}}^{w_{\text{rx}}}(\tau,\nu) = \sqrt{\frac{B\tau_p}{T}}\sum_{q=-\infty}^{\infty}e^{-j2\pi q\tau_p\nu}\text{rect}\left(\frac{\tau+q\tau_p}{T}\right) \\ \int_{\tau'}\text{sinc}(B\tau')n(\tau-\tau'+q\tau_p)d\tau'. \quad (65)$$

A general expression for the noise covariance for an arbitrary pulse-shaping filter in the matched filtering configuration is presented in [12] (see the expression after Eq. (47) in page 4469 of [12]). Here, for the case of sinc filter, using the DD domain noise in (65), we write the covariance of the noise as

$$A(\tau_1,\nu_1,\tau_2,\nu_2) = \mathbb{E}[n_{\text{dd}}^{w_{\text{rx}}}(\tau_1,\nu_1)n_{\text{dd}}^{w_{\text{rx}}*}(\tau_2,\nu_2)], \quad (66)$$

which can be written as (67) given at the top of this page. Using the covariance property of the noise that $\mathbb{E}[n(\tau_1)n^*(\tau_2)] = N_0\delta(\tau_2-\tau_1)$, where $\delta(\cdot)$ is the Kronecker delta function, the closed-form expression for noise covariance is obtained as (68) given at the top of this page, where $\lfloor\cdot\rfloor$ and $\lceil\cdot\rceil$ denote floor and ceiling operators, respectively.

Now, the covariance of the filtered noise samples given by (18) for the DOS system can be obtained using (68) given at the top of the next page, where $\mathbf{R}_{\text{dos}}[k_1NQ_\nu+l_1, k_2NQ_\nu+l_2] = A(\tau_1,\nu_1,\tau_2,\nu_2)|_{\tau_1=k_1\frac{\Delta\tau}{Q_\tau},\nu_1=l_1\frac{\Delta\nu}{Q_\nu},\tau_2=k_2\frac{\Delta\tau}{Q_\tau},\nu_2=l_2\frac{\Delta\nu}{Q_\nu}}$. For the PBS system, the covariance of the noise samples of the q'th branch is obtained using (68) as $\mathbf{R}^{(q')}[k_1N+l_1,k_2N+l_2] = A(\tau_1,\nu_1,\tau_2,\nu_2)$ when $\tau_1=(k_1+\frac{q_\tau}{Q_\tau})\Delta\tau$, $\nu_1=(l_1+\frac{q_\nu}{Q_\nu})\Delta\nu$, $\tau_2=(k_2+\frac{q_\tau}{Q_\tau})\Delta\tau$, and $\nu_2=(l_2+\frac{q_\nu}{Q_\nu})\Delta\nu$.

References

[1] R. Hadani et al., "Orthogonal time frequency space modulation," *Proc. IEEE WCNC'2017*, pp. 1-6, Mar. 2017.
[2] Z. Wei et al., "Orthogonal time-frequency space modulation: A promising next-generation waveform," *IEEE Wireless Commun. Mag.*, vol. 28, no. 4, pp. 136-144, Aug. 2021.
[3] Y. Hong, T. Thaj, and E. Viterbo, *Delay-Doppler Communications: Principles and Applications*, Academic Press, 2022.
[4] A. J. E. M. Janssen, "The Zak transform: A signal transform for sampled time-continuous signals," *Philips J. Res.*, 43, pp. 23-69, 1988.
[5] A. J. E. M. Janssen, "Bargmann transform, Zak transform, and coherent states," *J. Math. Phys.* 23, 720–731 (1982).
[6] S. K. Mohammed, R. Hadani, A. Chockalingam, and R. Calderbank, "OTFS - A mathematical foundation for communication and radar sensing in the delay-Doppler domain," *IEEE BITS the Inform. Theory Mag.*, vol. 2, no. 2, pp. 36-55, Nov. 2022.
[7] S. K. Mohammed, R. Hadani, A. Chockalingam, and R. Calderbank, "OTFS - Predictability in the delay-Doppler domain and its value to communication and radar sensing," *IEEE BITS the Inform. Theory Mag.*, vol. 3, no. 2, pp. 7-31, Jun. 2023.
[8] M. Ubadah, S. K. Mohammed, R. Hadani, S. Kons, A. Chockalingam, and R. Calderbank, "Zak-OTFS for integration of sensing and communication," online arxiv.org/abs/2404.04182, 5 Apr 2024.
[9] J. Jayachandran, R. K. Jaiswal, S. K. Mohammed, R. Hadani, A. Chockalingam, and R. Calderbank, "Zak-OTFS: Pulse shaping and the tradeoff between time/bandwidth expansion and predictability," online arxiv.org/abs/2405.02718, 4 May 2024.
[10] F. Jesbin and A. Chockalingam, "Near-optimal detection of Zak-OTFS signals," *IEEE ICC'2024*, pp. 4476-4481, Jun. 2024.
[11] S. Gopalam, I. B. Collings, S. V. Hanly, H. Inaltekin, S. R. B. Pillai, and P. Whiting, "Zak-OTFS implementation via time and frequency windowing," *IEEE Trans. Commun.*, vol. 72, pp. 3873-3889, Jul. 2024.
[12] S. Gopalam, H. Inaltekin, I. B. Collings, and S. V. Hanly, "Optimal Zak-OTFS receiver and its relation to the radar matched filter," *IEEE Open J. of the Commun. Soc.*, vol. 5, pp. 4462-4482, 2024.
[13] B. Dabak, V. Khammammetti, S. K. Mohammed, and R. Calderbank, "Zak-OTFS and LDPC codes," *Proc. IEEE ICC'2024*, pp. 3785-3790, Jun. 2024.
[14] J. Jayachandran, M. Ubadah, S. K. Mohammed, R. Hadani, and A. Chockalingam, "Zak-OTFS and turbo signal processing for joint sensing and communication," online: arXiv:2406.06024 10 Jun 2024.
[15] J. Jayachandran, I. A. Khan, S. K. Mohammed, R. Hadani, and A. Chockalingam, "Zak-OTFS with interleaved pilots to extend the region of predictable operation," online: arXiv:2408.09379 18 Aug 2024.
[16] Y. Ge, Q. Deng, P. C. Ching and Z. Ding, "Receiver design for OTFS with a fractionally spaced sampling approach", *IEEE Trans. Wireless Commun.*, vol. 20, no. 7, pp. 4072-4086, July 2021.
[17] P. Priya, E. Viterbo and Y. Hong, "Low complexity MRC detection for OTFS receiver with oversampling," *IEEE Trans. Wireless Commun.*, vol. 23, no. 2, pp. 1459-1473, Feb. 2024.
[18] N. Deconda, N. Seshadri, R. D. Koilpillai, "A modified MRC receiver for OTFS systems with oversampling," *IEEE WCNC'2024*, Apr. 2024.
[19] S. K. Mohammed, R. Hadani, and A. Chockalingam, *OTFS Modulation – Theory and Applications*, IEEE-Wiley, 2024.
[20] I. A. Khan and S. K. Mohammed, "A low-complexity OTFS channel estimation method for fractional delay-Doppler scenarios," *IEEE Wireless Commun. Lett.*, vol. 12, no. 9, pp. 1484-1488, Sep. 2023.

$$A(\tau_1,\nu_1,\tau_2,\nu_2) = \frac{N_0\tau_p}{T} \sum_{q_1=\left\lceil -\frac{N}{2}-\frac{\tau_1}{\tau_p}\right\rceil}^{\left\lfloor \frac{N}{2}-\frac{\tau_1}{\tau_p}\right\rfloor} \sum_{q_2=\left\lceil -\frac{N}{2}-\frac{\tau_2}{\tau_p}\right\rceil}^{\left\lfloor \frac{N}{2}-\frac{\tau_2}{\tau_p}\right\rfloor} e^{j2\pi(q_2\nu_2-q_1\nu_1)\tau_p}\,\mathrm{sinc}(B((\tau_2-\tau_1)+(q_2-q_1)\tau_p)). \tag{68}$$

[21] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge, U.K.: Cambridge Univ. Press, 2005.

[22] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge Univ. Press, 2013.

[23] ITU-R M.1225, “Guidelines for evaluation of radio transmission technologies for IMT-2000,” Int. Telecommun. Union Radio Commun., 1997.