Message passing-based sparse spatial channel estimation robust to partially coherent phase noise

Hamed Masoumi, Nitin Jonathan Myers
Delft Center for Systems and Control (DCSC)
Delft University of Technology, Delft, The Netherlands
Email: {H.Masoumi, N.J.Myers}@tudelft.nl

2025 IEEE 101st Vehicular Technology Conference (VTC2025-Spring) | 979-8-3315-3147-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/VTC2025-SPRING65109.2025.11174616

***Abstract*—Channel estimation can lead to a substantial training overhead in millimeter wave (mmWave) and terahertz (THz) systems employing large arrays. Prior work has leveraged channel sparsity at these frequencies to reduce this overhead. Most of the sparsity-aware algorithms, however, assume perfect phase coherence in the channel measurements, which is disrupted due to phase noise. Due to the errors induced by phase noise, standard sparse channel estimation algorithms assuming perfect phase coherence can fail. In this paper, we consider a frame structure in which the channel measurements are acquired over multiple packets. Our model assumes that the phase errors remain constant within a packet and vary considerably across different packets, leading to partially coherent channel measurements. We develop a message passing-based technique for sparse channel estimation under such partially coherent phase errors and show that our approach achieves a lower channel reconstruction error than comparable benchmarks.**

***Index Terms*—mm-Wave, terahertz, compressed sensing, model mismatch, phase errors**

I. Introduction

The high scattering at mmWave and THz frequencies results in sparse angle domain channels [1]. Compressed sensing (CS) exploits this sparse structure to estimate channels from sub-Nyquist measurements. Typically, the channel measurements are acquired sequentially over time and they are perturbed by phase errors due to jitter at the oscillators [2]. When the phase errors vary substantially over the collection of channel measurements, standard CS-based channel estimation methods [3], [4] fail due to a model mismatch. This issue is exacerbated in mmWave and THz bands, where phase noise is higher compared to lower frequency bands [5], [6].

We consider an IEEE 802.11ad/ay type frame structure in which the spatial channel measurements may be acquired over multiple packets. While the phase errors in the measurements within a packet are similar, the errors can vary significantly across different packets [7]. The sparse channel measurements acquired under such errors are called partially coherent CS measurements [8]. Prior work in [9]–[11] addressed this challenge by using only the magnitude of the CS measurements for sparse recovery. Such an approach, however, fails to exploit the partially coherent structure in the phase errors for sparse recovery. In [12], an extended Kalman filter is used within a standard CS algorithm to track the phase variations and acquire a one-sparse estimate for the channel. The first-order Taylor approximation of the phase error used in extended Kalman filtering provides a poor approximation of the exact errors at a high phase noise variance. Finally, the unknown phase errors and the sparse signal can be estimated using the lifting technique in [13]. The lifting-based approach demands a high complexity as it solves for a lifted signal whose dimension is much larger than the underlying sparse signal [7].

In [7], [8], [14], techniques that account for partial phase coherence in the CS measurements were developed. In [8], a two-stage partially coherent compressive phase retrieval (PC-CPR) was proposed for sparse channel estimation. The algorithm is initialized using the CPR method from [15], and the phase offsets and the sparse channel are iteratively estimated in a refinement stage. In [14], expectation-maximization generalized approximate message passing (EM-GAMP) [16] is used to first obtain a coarse estimation of the sparse channel, which is subsequently used to estimate the phase offsets. In [7], phase error-aware matching pursuit followed by alternating optimization was developed to recover the sparse channel and the phase errors. The methods in [6]–[8] assume that the channel is exactly sparse, which is unrealistic. Furthermore, [7], [14] assume that the sparsity level is known and [8] assumes that a bound on the number of paths in the channel is known, both of which limit their practical use.

In this paper, we develop a sparse channel estimation method that exploits the partially coherent structure in the measurements. In our approach, the phase errors are first absorbed into the sparse vector to construct multiple correlated sparse vectors that share the same support. The shared support structure is exploited using the multiple measurement vector approximate message passing (MMV-AMP) algorithm [17] to reconstruct the correlated sparse vectors. Finally, we recover the sparse channel and the phase errors from the set of correlated vectors by alternating optimization.

Notation: We use a, $\mathbf{a}$ and $\mathbf{A}$ to denote a scalar, vector, and a matrix. We use $(\cdot)^{\mathrm{T}}$and $(\cdot)^*$ to denote the transpose and conjugate-transpose operators. The i^{th} entry of $\mathbf{a}$ is indicated by $a[i]$. We denote the $(i,j)^{\mathrm{th}}$ entry of $\mathbf{A}$ by $A(i,j)$, the i^{th} row of $\mathbf{A}$ by $\mathbf{A}(i,:)$ and its j^{th} column by $\mathbf{A}(:,j)$. We use $\|\mathbf{A}\|_{\mathrm{F}}$ to denote the Frobenius norm of $\mathbf{A}$. $\mathrm{vec}(\mathbf{A})$ is the vector version of $\mathbf{A}$ obtained by stacking its columns. We use $\otimes$ to denote the Kronecker product. The inner product of $\mathbf{A}$ and $\mathbf{B}$ is $\langle \mathbf{A}, \mathbf{B} \rangle = \sum_{i,j} A(i,j)B^*(i,j)$. We use x for a random variable and x for its realization, with $p(x)$ as the PDF of x evaluated at x. The complex Gaussian PDF $\mathcal{CN}(x; a, b)$ has mean a and variance b. Finally, $\mathrm{j} = \sqrt{-1}$.

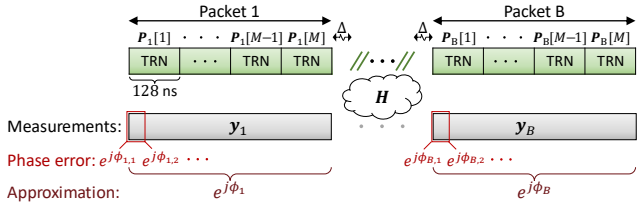

Fig. 1. The frame structure used to acquire partially coherent channel measurements. Our model approximates the phase errors to a constant within each packet. The phase errors can vary across packets.

II. System Model

We consider an $N \times N$ half-wavelength spaced uniform planar phased array (UPA) at the transmitter (TX) and a single antenna receiver (RX). The N^2 dimensional spatial channel between the TX and RX is modeled as an $N \times N$ matrix $\mathbf{H}$ to exploit the 2D structure in the channel. Let L denote the number of propagation rays in the environment with the ℓ^{th} ray having a complex gain of β_ℓ, an azimuth angle-of-departure (AoD) $\theta_{a,\ell}$ and an elevation AoD $\theta_{e,\ell}$. By defining the beamspace angles as $\omega_{a,\ell} = \pi \sin\theta_{e,\ell} \sin\theta_{a,\ell}$, $\omega_{e,\ell} = \pi \sin\theta_{e,\ell} \cos\theta_{a,\ell}$ [18] and the $N \times 1$ Vandermonde vector $\mathbf{a}_N(\omega)$ as

$$\mathbf{a}_N(\omega) = \left[1, e^{\mathrm{j}\omega}, e^{\mathrm{j}2\omega}, \cdots, e^{\mathrm{j}(N-1)\omega}\right]^{\mathrm{T}}, \tag{1}$$

the baseband channel matrix $\mathbf{H}$ is given by

$$\mathbf{H} = \sum_{\ell=1}^{L} \beta_\ell \mathbf{a}_N\left(\omega_{e,\ell}\right) \mathbf{a}_N^{\mathrm{T}}\left(\omega_{a,\ell}\right). \tag{2}$$

The channel dimension N^2 with typical mmWave or THz phased arrays can be in the order of hundreds to thousands.

The high scattering at mmWave and THz frequencies results in an approximately sparse channel in the angle domain [19]. To exploit a sparse prior in channel reconstruction, $\mathbf{H}$ is transformed into the angle domain. Since we assume a UPA at the TX, the 2D-DFT of $\mathbf{H}$ is used for its angle-domain representation. Let $\mathbf{U}_N$ denote the standard $N \times N$ unitary DFT matrix and $\mathbf{X}$ denote the angle-domain representation of $\mathbf{H}$. Then,

$$\mathbf{H} = \mathbf{U}_N \mathbf{X} \mathbf{U}_N, \tag{3}$$

where $\mathbf{X}$ is approximately sparse in typical mmWave or THz systems with large arrays.

We use the frame structure shown in Fig. 1, inspired by the signaling structure in IEEE 802.11ad/ay [20]. Here, the measurements of the unknown $\mathbf{H}$ are acquired over B training packets. We use M to denote the number of spatial channel measurements acquired in each packet. The RX acquires one spatial measurement of the channel when the TX applies a beamforming matrix to its array. Due to phase noise at the TX and the RX oscillators, the phase of the channel measurement is corrupted by an unknown phase error which is usually modeled as a Wiener process. This process is characterized by the power spectral density at an offset f_{m} from the carrier

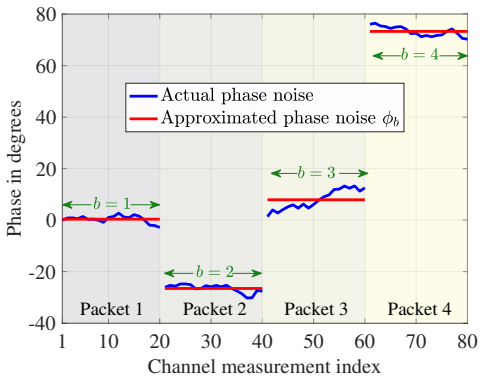

Fig. 2. A realization of phase noise and its approximation for $f_{\mathrm{c}} = 60$ GHz and $\mathcal{L}(f_{\mathrm{m}}) = -99.4$ dBc/Hz [21] at $f_{\mathrm{m}} = 1$ MHz.. Here, $M = 20$ measurements are acquired in each of the $B = 4$ packets.

frequency f_{c} in dBc/Hz. We denote this density as $\mathcal{L}\left(f_{\mathrm{m}}\right)$ [2] and it is given by

$$\mathcal{L}(f_{\mathrm{m}}) = 10 \log_{10}\left(\frac{f_{\mathrm{c}}^2 c}{\pi^2 f_{\mathrm{c}}^4 c^2 + f_{\mathrm{m}}^2}\right) \text{ dBc/Hz}, \tag{4}$$

where c is an oscillator-dependent constant. An oscillator with a small $\mathcal{L}(f_{\mathrm{m}})$ results in phase errors with a low variance.

For a Wiener process, the variance of the phase error, conditioned on a past phase error sample, increases with the time lapsed. Let Δ denote the time lag between successive packets. Since Δ is typically large (e.g., 44 μs) compared to the time needed to acquire each measurement (e.g., 128 ns) [7], [14], the phase variations within each packet can be ignored. The phase offsets across different packets, however, are significant due to the large time lapse and cannot be ignored, as seen from Fig. 2. We model only inter-packet phase errors to develop our algorithm, but we evaluate our method using realistic phase errors that also vary within each packet.

We use the scalar ϕ_b to model the phase error for the measurements obtained in packet b. For a Wiener phase noise process, $\phi_b | \phi_{b-1} \sim \mathcal{N}\left(\phi_{b-1}, \sigma_{\mathrm{pn}}^2\right)$, where the variance $\sigma_{\mathrm{pn}}^2 = 4\pi^2 f_{\mathrm{c}}^2 c \Delta$. Here, c is the phase noise parameter determined from $\mathcal{L}\left(f_{\mathrm{m}}\right)$ and f_{c} is the carrier frequency [2]. Without loss of generality, we assume $\phi_1 = 0$. In practice, the nonzero ϕ_1 is absorbed into $\mathbf{X}$ as it usually suffices to estimate $\mathbf{X}$ up to a global phase ambiguity. In Fig. 2, a realization of the phase noise and its approximation ϕ_b are shown. We use $\mathbf{P}_b[m] \in \mathbb{C}^{N \times N}$ to denote the beamformer applied at the TX in b^{th} packet, for the RX to obtain the m^{th} spatial channel measurement denoted by $y_b[m]$. For a measurement noise $w_b[m] \sim \mathcal{CN}(0, \sigma^2)$, the m^{th} spatial channel measurement acquired in packet b is

$$y_b[m] = e^{\mathrm{j}\phi_b} \langle \mathbf{H}, \mathbf{P}_b[m] \rangle + w_b[m] \quad \forall m, \forall b. \tag{5}$$

Our goal is to estimate $\mathbf{H}$, from MB phase perturbed measurements in (5), by exploiting its angle-domain sparsity.

Now, we discuss the partially coherent CS formulation from [8]. We define $\mathbf{A}_b \in \mathbb{C}^{M \times N^2}$ as the CS matrix associated with M spatial measurements from the b^{th} packet in (5). The

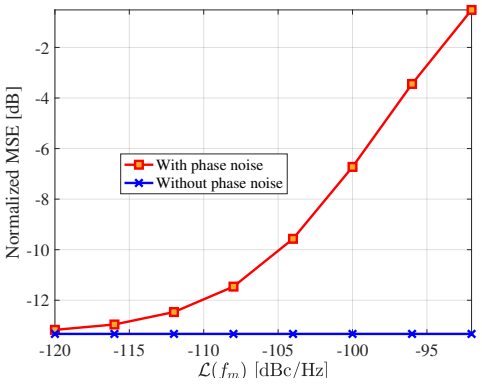

Fig. 3. Standard sparse channel estimation using [22] fails when the measurements are perturbed by phase noise at $f_c = 60$ GHz and $\mathcal{L}(f_m)$ dBc/Hz are at $f_m = 1$ MHz. Here, $M = 64$ measurements of a 16×16 channel were acquired in each of the $B = 4$ packets.

m^{th} row of $\mathbf{A}_b$ is $\text{vec}\,(\mathbf{P}_b^*[m])\,(\mathbf{U}_N \otimes \mathbf{U}_N)$. In the b^{th} packet, let $\mathbf{y}_b$ denote the vector of M measurements $\{y_b[m]\}_{m=1}^M$ in (5) and $\mathbf{w}_b$ contain the corresponding measurement noise $\{w_b[m]\}_{m=1}^M$. Also, we use the N^2-dimensional sparse vector $\mathbf{x}$ to represent $\mathbf{X}$ in the vector form, i.e., $\mathbf{x} = \text{vec}\,(\mathbf{X})$. Now, we can write $\mathbf{y}_b = [y_b[1], \cdots y_b[M]]^{\text{T}}$ as

$$\mathbf{y}_b = e^{\text{j}\phi_b}\mathbf{A}_b\mathbf{x} + \mathbf{w}_b. \tag{6}$$

In practice, the measurements $\{\mathbf{y}_b\}_{b=1}^B$ are acquired at the RX and the CS matrices $\{\mathbf{A}_b\}_{b=1}^B$ are known. The phase errors $\{\phi_b\}_{b=2}^B$ and the sparse vector $\mathbf{x}$ are unknown and have to be estimated from the acquired measurements. We observe from Fig. 3 that standard sparse recovery [22] which is agnostic to ϕ_bs fails to recover the sparse channel from its phase perturbed measurements in (6). The reconstruction is poor at large $\mathcal{L}(f_m)$, i.e., when the phase noise is high.

III. Proposed Channel Estimation Method

Our method absorbs the phase offsets $\{e^{\text{j}\phi_b}\}_{b=1}^B$ into the sparse vector $\mathbf{x}$ to define the unknown vector

$$\mathbf{z}_b = e^{\text{j}\phi_b}\mathbf{x}. \tag{7}$$

As $\mathbf{z}_b$ is just a scalar multiple of $\mathbf{x}$, $\mathbf{z}_b$ is sparse when $\mathbf{x}$ is sparse. Let $\mathcal{S}_\mathbf{x}$ denote the support of $\mathbf{x}$, i.e., $\mathcal{S}_\mathbf{x} \triangleq \{n : x[n] \neq 0\}$, and $\mathcal{S}_{\mathbf{z}_b}$ denote the support of $\mathbf{z}_b$. From (7), we observe that $\mathbf{z}_b$ has the same support as $\mathbf{x}$ for each b, i.e., $\mathcal{S}_{\mathbf{z}_1} = \mathcal{S}_{\mathbf{z}_2} = \cdots = \mathcal{S}_{\mathbf{z}_B} = \mathcal{S}_\mathbf{x}$. Substituting (7) in (6) yields

$$\mathbf{y}_b = \mathbf{A}_b\mathbf{z}_b + \mathbf{w}_b. \tag{8}$$

By absorbing the phase errors into the sparse vector, the measurement model in (8) is a linear measurement model in the unknown vectors $\{\mathbf{z}_b\}_{b=1}^B$. Further, the vectors $\{\mathbf{z}_b\}_{b=1}^B$ are sparse and they all share a common support, i.e., they are non-zero at the same indices.

Our approach first reconstructs $\{\mathbf{z}_b\}_{b=1}^B$ by exploiting sparsity and the common support structure using the MMV-AMP algorithm [17]. To explain how MMV-AMP is used, we model $\mathbf{z}_b$ as a realization of a random vector $\mathbf{z}_b$ and $\mathbf{x}$ as a realization of a random vector $\mathbf{x}$. We define $\mathbf{s} \in \{0,1\}^{N^2}$ as the N^2-dimensional random binary vector, with independent and identically distributed (IID) entries, to model the support of $\{\mathbf{z}_b\}_{b=1}^B$. We use λ to denote the probability that $\mathsf{x}[n]$ is non-zero, which is also the probability that $\mathsf{z}_b[n]$ is non-zero. Then,

$$p(s[n]) = \lambda^{s[n]}\,(1-\lambda)^{1-s[n]}, \quad s[n] \in \{0,1\}. \tag{9}$$

The amplitude of the random vectors $\{\mathbf{z}_b\}_{b=1}^B$ is modeled using $\{\mathbf{r}_b\}_{b=1}^B$. We assume that the entries of $\{\mathbf{r}_b\}_{b=1}^B$ are IID as a zero mean Gaussian distribution with variance γ, i.e.,

$$p(r_b[n]) = \mathcal{CN}(r_b[n]; 0, \gamma)\,. \tag{10}$$

Now, $\mathsf{z}_b[n]$ can be decomposed as the product of the amplitude variable and the support variable as $\mathsf{z}_b[n] = \mathsf{s}[n]\mathsf{r}_b[n]$. The conditional probability of $\mathsf{z}_b[n] = z_b[n]$ is given by

$$p\Big(z_b[n]\Big|s[n], r_b[n]\Big) = \delta(z_b[n] - s[n]r_b[n])\,. \tag{11}$$

From (11), we observe that $z_b[n] = 0$ when $s[n] = 0$, thereby aiding sparse priors. The shared support structure across $\{\mathbf{z}_b\}_{b=1}^B$ is inherently incorporated as the supports of $\{\mathbf{z}_b\}_{b=1}^B$ are modeled by a single random vector $\mathbf{s}$.

Next, the known dependencies between $\{\mathbf{z}_b\}_{b=1}^B$, $\mathbf{s}$ and $\{\mathbf{r}_b\}_{b=1}^B$ leveraged to factorize their posterior joint PDF as

$$p\Big(\{\mathbf{z}_b\}_{b=1}^B, \mathbf{s}, \{\mathbf{r}_b\}_{b=1}^B, \Big|\{\mathbf{y}_b\}_{b=1}^B\Big) \propto \prod_{b=1}^B \Bigg(\prod_{m=1}^M p\Big(y_b[m]\Big|\mathbf{z}_b\Big) \prod_{n=1}^N p\Big(z_b[n]\Big|s[n], r_b[n]\Big)\, p\,(r_b[n])\Bigg) \prod_{n=1}^N p\,(s[n])\,, \tag{12}$$

where $\propto$ denotes equality up to a constant scale factor and $p(y_b[m]\,|\mathbf{z}_b\,)$ denotes the likelihood of $y_b[m]$ when $\mathbf{z}_b = \mathbf{z}_b$ is given. See Table I for the factors in (12).

TABLE I
Factors of the joint PDF in (12).

Factor definition	Explicit form
$p\Big(y_b[m]\Big\|\mathbf{z}_b\Big)$	$\mathcal{CN}(y_b[m]; \mathbf{A}_b(m,:)\mathbf{z}_b, \sigma^2)$
$f_b[n] = p\Big(z_b[n]\Big\|s[n], r_b[n]\Big)$	$\delta(z_b[n] - s[n]r_b[n])$
$p(s[n])$	$\lambda^{s[n]}\,(1-\lambda)^{1-s[n]}\,,\ s[n] \in \{0,1\}$
$p(r_b[n])$	$\mathcal{CN}(r_b[n]; 0, \gamma)$

The dependencies among different variables in (12), shown in Fig. 4, is leveraged for sparse recovery of $\{\mathbf{z}_b\}_{b=1}^B$. The factor graph in Fig. 4 comprises two types of nodes. The nodes shown as rectangles are called *factor nodes*, each corresponding to a factor in (12). The nodes shown as circles are called *variable nodes*. In a factor graph, each factor node is linked to variable nodes that comprise its argument. For example, the factor node corresponding to $p\Big(z_b[n]\Big|s[n], r_b[n]\Big)$ is connected to nodes associated with the variables $\mathsf{r}_b[n]$, $\mathsf{s}[n]$ and $\mathsf{z}_b[n]$. The factor graph in Fig. 4 consists of B planes, where plane b corresponds to packet b that acquires measurements in (8). The

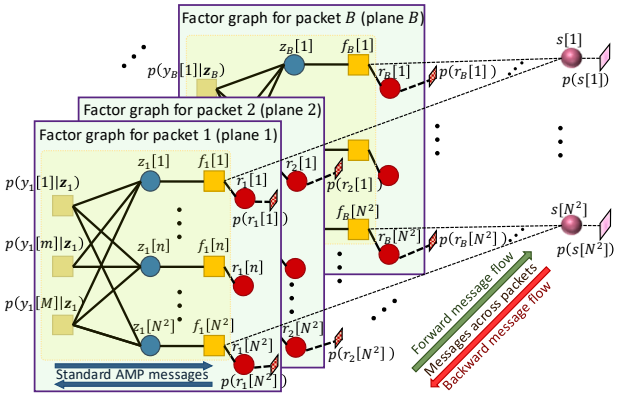

Fig. 4. The factor graph we use to compute MMSE estimates of $\{\mathbf{z}_b\}_{b=1}^B$ given all the measurements $\{\mathbf{y}_b\}_{b=1}^B$ where $f_b[n] = p(z_b[n]\,|s[n], r_b[n]\,)$ given in (11).

connections between different planes capture the correlations among $\{\mathbf{z}_b\}_{b=1}^B$ which follows from (7). For instance, the n^{th} factor nodes $\{f_b[n]\}_{b=1}^B$ across all planes are connected via $\mathsf{s}[n]$ in Fig. 4. This is because $\{\mathsf{z}_b[n]\}_{b=1}^B$ are jointly non-zero when $s[n] = 1$.

Our approach aims to find the minimum mean squared error (MMSE) estimates of $\{\mathbf{z}_b\}_{b=1}^B$ given the measurements $\{\mathbf{y}_b\}_{b=1}^B$ using factor graph in Fig. 4. To this end, the marginal posteriors $p\left(z_b[n]\middle|\{\mathbf{y}_b\}_{b=1}^B\right)$ are first estimated by employing the sum-product algorithm [23] for the factor graph in Fig. 4 which results in the MMV-AMP algorithm [17]. The means of these posteriors then yield the MMSE estimates.

To explain how the MMV-AMP algorithm [17] is used to estimate the marginal posteriors $p\left(z_b[n]\middle|\{\mathbf{y}_b\}_{b=1}^B\right)$ from our factor graph in Fig. 4, we first briefly describe the sum-product algorithm [23]. In each iteration of the sum-product algorithm, nodes in the factor graph exchange messages, also known as beliefs, which represent probability densities. There are two types of message flows: those sent from variables to neighboring factors, and those sent from factors to neighboring variables. A belief exchanged from a factor to a variable represents the probability distribution of the variable, as perceived from the factor's perspective. In contrast, the belief sent from a variable to a factor represents the variable's perception of its own probability distribution, informed by all the factor nodes it is connected to—except for the one it is messaging. In the sum-product algorithm [23], a message from a variable to a factor is obtained by multiplying all incoming messages from the other neighboring factors. A message from a factor to a variable is calculated by summing (or integrating) the product of all incoming messages over all variables, except for the one to which the message is sent.

We now briefly explain message propagation in the MMV-AMP. The messages are propagated on a plane-by-plane basis to exploit the shared support. In the first plane ($b = 1$) of Fig. 4, the linear regression problem in (8) is solved with a sparse prior on $\mathbf{z}_b$. To this end, the standard approximate message passing (AMP) [22] is used to solve the linear regression problem in (8). In this step, a sparse prior on the channel is incorporated through the messages from $\{f_b[n]\}_{n=1}^{N^2}$ and faithfulness to measurements through $p\left(y_b[m]\middle|\mathbf{z}_b\right)$, $\forall m = \{1, \cdots, M\}$. After convergence in this step, the marginal posterior of $\mathsf{z}_1[n]$ is estimated for each n. This posterior provides side information about the support of $\mathsf{z}_2[n]$, a variable to be solved in the next plane ($b = 2$). Next, this side information, along with the priors $p(r_2[n])$ and $p(s[n])$ $\forall n = \{1, \cdots, N^2\}$ are combined according to the sum-product rule which is then used as prior to solving for the posterior of $\mathbf{z}_2$ in (8) using the AMP. The procedure is continued until an estimate of the marginal posterior PDF for $\mathsf{z}_B[n]$ in the last plane, i.e., plane B, is obtained. Because the messages are passed in the direction of the increasing plane index b, we refer to this process as the *forward* message flow as shown in Fig. 4. The forward flow can sequentially update the channel estimate with each new measurement packet.

When the measurements from all the B packets are available for channel estimation, *backward* message flows can also be performed in addition to the forward message flows [17]. The backward flow propagates side information from plane $b+1$ to plane b, i.e., measurements in the $b+1^{\text{th}}$ packet are used to derive priors needed to solve for $\mathsf{z}_b[n]$. After multiple forward-backward message flows, the marginal posterior $p(z_b[n]\big|\{\mathbf{y}_b\}_{b=1}^B)$ is estimated by exploiting measurements from all the packets. The mean of these posteriors gives the estimates $\{\hat{\mathbf{z}}_b\}_{b=1}^B$.

We now discuss our alternating optimization method to recover $\mathbf{x}$ from the estimates $\{\hat{\mathbf{z}}_b\}_{b=1}^B$. To this end, we stack the estimates $\{\hat{\mathbf{z}}_b\}_{b=1}^B$ in an $N^2 \times B$ matrix $\hat{\mathbf{Z}} = [\hat{\mathbf{z}}_1, ..., \hat{\mathbf{z}}_B]$. We define $\boldsymbol{\phi} = [\phi_1, ..., \phi_B]^{\mathrm{T}}$, $e^{\mathrm{j}\boldsymbol{\phi}} = \left[e^{\mathrm{j}\phi_1}, ..., e^{\mathrm{j}\phi_B}\right]^{\mathrm{T}}$ and $\mathbf{Z} = [\mathbf{z}_1, ..., \mathbf{z}_B]$. From (7), we observe that $\mathbf{Z}$ is a rank one matrix as $\mathbf{Z} = \mathbf{x}e^{\mathrm{j}\boldsymbol{\phi}^{\mathrm{T}}}$. Hence, we find a rank-one matrix of the form $\mathbf{x}e^{\mathrm{j}\boldsymbol{\phi}^{\mathrm{T}}}$ that is closest to $\hat{\mathbf{Z}}$, i.e.,

$$\{\hat{\mathbf{x}}, \hat{\boldsymbol{\phi}}\} = \underset{\{\mathbf{x},\boldsymbol{\phi}\}}{\operatorname{argmin}} \ \|\hat{\mathbf{Z}} - \mathbf{x}e^{\mathrm{j}\boldsymbol{\phi}^{\mathrm{T}}}\|_{\mathrm{F}}^2, \tag{13}$$

where $\hat{\mathbf{x}}$ denotes the estimated $\mathbf{x}$ and $\hat{\boldsymbol{\phi}}$ is the estimated $\boldsymbol{\phi}$. To solve (13), we develop an alternating approach outlined in Algorithm 1. We use $\hat{\mathbf{x}}^{(i)}$ and $\hat{\boldsymbol{\phi}}^{(i)}$ to denote the estimates of $\mathbf{x}$ and $\boldsymbol{\phi}$ in the i^{th} iteration. $\hat{\boldsymbol{\phi}}^{(i)}$ is computed by solving (13) for a fixed $\mathbf{x}$ given by $\hat{\mathbf{x}}_b^{(i)}$. In this step, $\hat{\boldsymbol{\phi}}^{(i)}$ is computed as

$$\begin{aligned}\hat{\boldsymbol{\phi}}^{(i)} &= \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \ \|\hat{\mathbf{Z}} - \hat{\mathbf{x}}^{(i)}e^{\mathrm{j}\boldsymbol{\phi}^{\mathrm{T}}}\|_{\mathrm{F}}^2 \\ &= \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \ \Re\left(\sum_{b=1}^{B} \hat{\mathbf{z}}_b^* \hat{\mathbf{x}}^{(i)} e^{\mathrm{j}\phi_b}\right) = -\angle\hat{\mathbf{Z}}^*\hat{\mathbf{x}}^{(i)}. \end{aligned} \tag{14}$$

In (14), the operator $\Re(\cdot)$ returns the real part, and $\angle(\cdot)$ returns the angle of a complex number. Next, $\hat{\mathbf{x}}^{(i+1)}$ is updated by solving (13) for a fixed $\boldsymbol{\phi}$ given by $\hat{\boldsymbol{\phi}}^{(i)}$ (14) from the previous iteration. This results in the optimization problem

$$\hat{\mathbf{x}}^{(i+1)} = \underset{\mathbf{x}}{\operatorname{argmin}} \ \|\hat{\mathbf{Z}} - \mathbf{x}e^{\mathrm{j}\left(\hat{\boldsymbol{\phi}}^{(i)}\right)^{\mathrm{T}}}\|_{\mathrm{F}}^2, \tag{15}$$

Algorithm 1: Alternating algorithm to solve (13).

Input: $\{\hat{\mathbf{z}}_b\}_{b=1}^{B}$, Maximum number of iterations $T_{\max}$.
Initialization: $i=1$, $\hat{\mathbf{x}}^{(1)}=\hat{\mathbf{z}}_1$.
For $i=1,\cdots,T_{\max}$ **do**:
1. $\hat{\boldsymbol{\phi}}^{(i)}=-\angle\hat{\mathbf{Z}}^{*}\hat{\mathbf{x}}^{(i)}$. Note: $\angle\mathbf{c}$ returns the angle of $\mathbf{c}$
2. $\hat{\mathbf{x}}^{(i+1)}=\frac{1}{B}\hat{\mathbf{Z}}e^{-\mathrm{j}\hat{\boldsymbol{\phi}}^{(i)}}$.

End For
Output: $\hat{\mathbf{x}}=\hat{\mathbf{x}}^{(T_{\max})}$.

which is a convex problem. Therefore, $\hat{\mathbf{x}}^{(i+1)}$ can be computed by setting the gradient of the objective function in (15) with respect to $\mathbf{x}$ equal to zero and then solving for $\mathbf{x}$, resulting in

$$\hat{\mathbf{x}}^{(i+1)}=\frac{1}{B}\hat{\mathbf{Z}}e^{-\mathrm{j}\hat{\boldsymbol{\phi}}^{(i)}}. \tag{16}$$

This procedure, outlined in Algorithm 1, is repeated until the maximum number of iterations, denoted by $T_{\max}$, is reached. The sparse estimate $\hat{\mathbf{x}}$ is reshaped to an $N\times N$ matrix, which is then used to find the channel estimate $\hat{\mathbf{X}}$ from (3).

IV. Simulation Results

We consider a narrowband system with a 16×16 UPA operating at a carrier frequency of $f_{\mathrm{c}}=60$ GHz. The TX-RX distance is 15 m. Phase noise at $f_{\mathrm{m}}=1$ MHz offset from f_{c} is $\mathcal{L}(f_{\mathrm{m}})=-99.4$ dBc/Hz [21]. The duration to obtain each measurement is 128 ns and the time lag between successive packets is $\Delta=44$ μs [7] resulting in $\sigma_{\mathrm{pn}}^2\approx 0.2$ [2]. We use $B=4$ packets. The beam training matrices $\{\mathbf{P}_b[m]\}_{m=1}^{M}$ $\forall b\in\{1,\cdots,B\}$ applied at the UPA are random circular shifts of a 16×16 perfect binary array [4]. For each beam training matrix applied at the TX, training sequences of duration 128 ns within the TRN fields in Fig. 1 are used for transmission [4], resulting in a spreading gain of about 20 dB. We use 100 urban micro line-of-sight channels from the NYU simulator [24]. Therefore, AoDs can be off-grid, unlike the work in [7] that assumes on-grid channels with known sparsity.

The channel dataset is used to compute the prior parameters λ and γ in (9) and (10). To compute the sparsity rate λ, we first find the number of entries in the angle-domain channels with the largest magnitude such that they account for at least 95 percent of the energy in the channel. Denoting this number by K, we then calculate λ as the mean of the ratio K/N^2 over all channels. Next, the sample variance of the K entries from all the channels is used to compute γ. This resulted in $\lambda\approx 0.0712$ and $\gamma\approx 13.3$.

We use $\hat{\mathbf{H}}$ to denote the estimated channel obtained from its angle-domain estimate $\hat{\mathbf{X}}$ by inverse 2D-DFT from (3). Since we assume $\phi_1=0$, there is no phase ambiguity in the estimate $\hat{\mathbf{H}}$ and we can define normalized mean squared error (NMSE) as $\mathbb{E}[\|\mathbf{H}-\hat{\mathbf{H}}\|_{\mathrm{F}}^2/\|\mathbf{H}\|_{\mathrm{F}}^2]$. Signal-to-noise ratio (SNR) is given by $\mathrm{SNR}=\sum_{b=1}^{B}\|\mathbf{A}_b\mathbf{x}\|^2/MB\sigma^2$. We use $T_{\max}=5$ in Algorithm 1.

We use *SparseLift* from [13] as one of the benchmark methods. *SparseLift* translates the unknowns $\mathbf{x}$ and $\boldsymbol{\phi}$ into a higher dimension as $\boldsymbol{\Gamma}=e^{\mathrm{j}\boldsymbol{\phi}}\mathbf{x}^{\mathrm{T}}$ [7]. The standard CS measurement model for the lifted problem can be obtained by defining $\mathbf{y}=[\mathbf{y}_1^{\mathrm{T}},\ldots,\mathbf{y}_B^{\mathrm{T}}]^{\mathrm{T}}$, $\mathbf{A}=[\mathbf{A}_1^{\mathrm{T}},\ldots,\mathbf{A}_B^{\mathrm{T}}]^{\mathrm{T}}$, $\mathbf{w}=[\mathbf{w}_1^{\mathrm{T}},\ldots,\mathbf{w}_B^{\mathrm{T}}]^{\mathrm{T}}$, and the $MB\times B$ matrix $\mathbf{C}=\mathbf{I}_B\otimes\mathbf{1}_M$ where $\mathbf{I}_B$ is the $B\times B$ identity matrix and $\mathbf{1}_M$ is the $M\times 1$ vector of all ones. For more details on the application of *SparseLift*, we refer the reader to [7]. Denoting the $MB\times N^2B$ CS matrix corresponding to the lifted problem using $\mathbf{A}_{\mathrm{lift}}$ with its i^{th} row given by $\mathbf{A}(i,:)\otimes\mathbf{C}(i,:)$, the CS measurement model for the lifted problem is $\mathbf{y}=\mathbf{A}_{\mathrm{lift}}\mathrm{vec}(\boldsymbol{\Gamma})+\mathbf{w}$. *SparseLift* estimates $\boldsymbol{\Gamma}$ from this model by exploiting its sparsity, but not the shared support structure as done in our approach. For a fair comparison, we use the AMP [22] to solve the problem in *SparseLift*. Finally, the singular-value decomposition of the estimate $\hat{\boldsymbol{\Gamma}}$ is used to determine $\hat{\mathbf{x}}$.

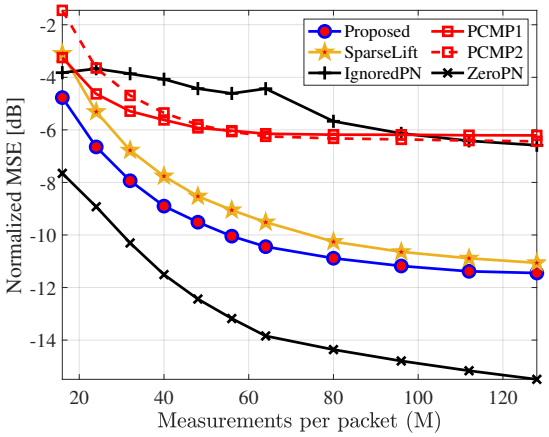

Fig. 5. NMSE with the number of CS measurements per packet for $B=4$, SNR $=-10$ dB. With phase perturbed CS measurements, the proposed method results in a smaller NMSE than benchmarks.

We also use the partially coherent matching pursuit (PCMP) algorithm from [7] to benchmark our method. PCMP is a greedy algorithm that iteratively detects the support of $\mathbf{x}$, similar to the matching pursuit algorithm. Then, the phase error and the coefficient corresponding to the detected support are estimated using an alternating optimization similar to Algorithm 1. PCMP requires knowledge of the sparsity level for $\mathbf{x}$. Therefore, we use PCMP algorithms with two different sparsity levels as benchmarks. PCMP1 uses a sparsity level of $\lceil\lambda N^2\rceil=19$ and PCMP2 uses a sparsity level of $2\lceil\lambda N^2\rceil=38$ with $\lceil\cdot\rceil$ denoting the ceil operator. We also consider two other benchmarks that both consider the standard linear measurement model and employ standard AMP [22] for sparse recovery. The first of these two benchmarks "IgnoredPN" uses phase perturbed measurements and suffers from a model mismatch, and the second 'ZeroPN" uses ideal phase error-free measurements.

From Fig. 5, we observe that our technique brings around 1 dB improvement in the NMSE compared to *SparseLift*. This is because our technique leverages the structure in partially coherent measurements such as the common support of the phase perturbed versions of the angle-domain channel in the factor graph in Fig. 4. The gap between our proposed technique and the *SparseLift*, however, is small at large M as

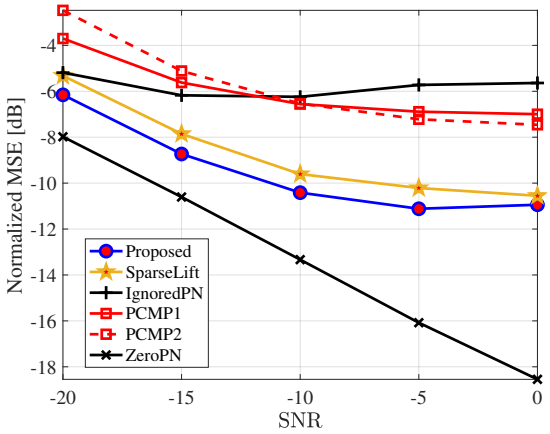

Fig. 6. NMSE with SNR for $B = 4$ and $M = 64$. With phase perturbed measurements, the proposed method results in a lower channel estimation error than the benchmarks.

likelihoods have a stronger impact on support identification than messages incorporating shared support structure. Next, we observe that there is almost no improvement with our method and SparseLift when $M \geq 110$. This is because our assumption that phase noise is almost the same over a packet breaks down for a long packet, equivalently a large M.

From Fig. 6, we observe that our technique achieves a lower NMSE than the *SparseLift*, especially at a low SNR. As SNR increases, NMSE decreases with both the proposed method and the *SparseLift*. The gap between these two methods also decreases. This is because increasing SNR does not help when phase noise becomes the limiting factor for sparse channel estimation. Finally, we observe that the NMSE of the IgnoredPN method remains almost unchanged by increasing SNR. This shows that the NMSE with this method is limited by phase errors. Finally, we observe that PCMP1 and PCMP2 do not perform well compared to the proposed method and the *SparseLift*. This is because PCMP is designed for exactly sparse channels with known sparsity levels, while the channels in our simulations are approximately sparse.

V. Conclusions

In this paper, we developed a message passing-based technique that allows sparse channel estimation under partially coherent phase noise. Our approach absorbs the phase errors into the sparse channel to define a collection of phase perturbed sparse vectors. The proposed method operates in two stages. First, it estimates the phase perturbed sparse channels by exploiting their common support structure and sparsity. Next, it employs an alternating optimization technique to reconstruct the angle-domain channel from the phase perturbed collection of sparse vectors. Numerical results demonstrate that our method achieves a lower mean squared error in channel recovery compared to recent benchmarks.

References

[1] R. W. Heath, N. Gonzalez-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 436–453, 2016.

[2] A. Demir, A. Mehrotra, and J. Roychowdhury, "Phase noise in oscillators: a unifying theory and numerical methods for characterization," *IEEE Trans. on Circuits and Syst. I: Fundamental Theory and Applications*, vol. 47, no. 5, pp. 655–674, 2000.

[3] A. Alkhateeb, G. Leus, and R. W. Heath, "Compressed sensing based multi-user millimeter wave systems: How many measurements are needed?" in *Proc. IEEE Intl. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2015, pp. 2909–2913.

[4] N. J. Myers, A. Mezghani, and R. W. Heath, "FALP: Fast beam alignment in mmWave systems with low-resolution phase shifters," *IEEE Trans. Commun.*, vol. 67, no. 12, pp. 8739–8753, 2019.

[5] Z. Sha and Z. Wang, "Channel estimation and equalization for terahertz receiver with RF impairments," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 6, pp. 1621–1635, 2021.

[6] R. Zhang, B. Shim, and H. Zhao, "Downlink compressive channel estimation with phase noise in massive MIMO systems," *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5534–5548, 2020.

[7] W. Yi, N. J. Myers, and G. Joseph, "Sparse millimeter wave channel estimation from partially coherent measurements," in *Proc. IEEE Intl. Conf. Commun. (ICC)*. IEEE, 2024, pp. 1897–1902.

[8] C. Hu, X. Wang, L. Dai, and J. Ma, "Partially coherent compressive phase retrieval for millimeter-wave massive MIMO channel estimation," *IEEE Trans. Signal Process.*, vol. 68, pp. 1673–1687, 2020.

[9] M. E. Rasekh and U. Madhow, "Noncoherent compressive channel estimation for mm-wave massive MIMO," in *Proc. Asilomar Conf. Signals Syst. Comput.* IEEE, 2018, pp. 889–894.

[10] X. Li, J. Fang, H. Duan, Z. Chen, and H. Li, "Fast beam alignment for millimeter wave communications: A sparse encoding and phaseless decoding approach," *IEEE Trans. Signal Process.*, vol. 67, no. 17, pp. 4402–4417, 2019.

[11] K.-H. Liu, X. Li, H. Zhao, and G. Fan, "Structured phase retrieval-aided channel estimation for millimeter-wave/sub-terahertz MIMO systems," in *Proc. IEEE Vehicular Tech. Conf. (VTC2022-Fall)*, 2022, pp. 1–5.

[12] H. Yan and D. Cabria, "Compressive sensing based initial beamforming training for massive MIMO millimeter-wave systems," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, 2016, pp. 620–624.

[13] S. Ling and T. Strohmer, "Self-calibration and biconvex compressive sensing," *Inverse Problems*, vol. 31, no. 11, p. 115002, 2015.

[14] C. Wei, H. Jiang, J. Dang, L. Wu, and H. Zhang, "Accurate channel estimation for mmwave massive MIMO with partially coherent phase offsets," *IEEE Commun. Lett.*, vol. 26, no. 9, pp. 2170–2174, 2022.

[15] L. Zhang, G. Wang, G. B. Giannakis, and J. Chen, "Compressive phase retrieval via reweighted amplitude flow," *IEEE Trans. Signal Process.*, vol. 66, no. 19, pp. 5029–5040, 2018.

[16] J. P. Vila and P. Schniter, "Expectation-maximization Gaussian-mixture approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4658–4672, 2013.

[17] J. Ziniel and P. Schniter, "Efficient high-dimensional inference in the multiple measurement vector problem," *IEEE Trans. Signal Process.*, vol. 61, no. 2, pp. 340–354, 2012.

[18] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge university press, 2005.

[19] H. Sarieddeen, M.-S. Alouini, and T. Y. Al-Naffouri, "An overview of signal processing techniques for terahertz communications," *Proc. IEEE*, vol. 109, no. 10, pp. 1628–1665, 2021.

[20] Y. Ghasempour, C. R. Da Silva, C. Cordeiro, and E. W. Knightly, "IEEE 802.11ay: Next-generation 60 GHz communication for 100 Gb/s Wi-Fi," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 186–192, 2017.

[21] J. Rimmelspacher, R. Ciocoveanu, G. Steffan, M. Bassi, and V. Issakov, "Low power low phase noise 60 GHz multichannel transceiver in 28 nm CMOS for radar applications," in *Proc. IEEE Radio Frequency Integrated Circuits Symposium (RFIC)*, 2020, pp. 19–22.

[22] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *IEEE Intl. Symposium on Information Theory Proceedings*. IEEE, 2011, pp. 2168–2172.

[23] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[24] S. Sun, G. R. MacCartney, and T. S. Rappaport, "A novel millimeter-wave channel simulator and applications for 5G wireless communications," in *Proc. IEEE Intl. Conf. Commun. (ICC)*, 2017, pp. 1–7.