ML5G-PHY [channel estimation] ITU AI/ML 5G Challenge A Sparse Bayesian Learning Solution

ICARUS: Özlem Tuğfe Demir, Cenk M. Yetis, Emil Björnson, Pontus Giselsson

***Abstract*—To tackle this challenging problem, we propose mainly a sparse Bayesian learning algorithm to exploit the sparsity of the channel. We utilize the pattern-coupling concept to model possible block sparsity patterns among the consecutive angle-of-arrivals (AoAs) and angle-of-departures (AoDs). As a first step, we obtain the time-domain channels from the provided training dataset by inverse FFT and remove the negligibly small taps. Then, we apply the algorithm to the true time-domain channels to obtain the sparse representations. Using joint angular distribution that is learned from training data, we refine the grids and pattern-coupling relations in testing stage in an aim to improve the channel estimation quality.**

I. System Model

We will follow the notation of [1] with some minor differences. We denote the number of delay taps by L and the ℓth delay tap of the channel is represented bt a $N_r \times N_t$ matrix denoted as $\mathbf{H}_\ell$. Each channel $\mathbf{H}_\ell$ can be expressed in compact form as follows

$$\mathbf{H}_\ell = \mathbf{A}_{\mathrm{R}} \mathbf{\Delta}_\ell \mathbf{A}_{\mathrm{T}}^{\mathrm{H}}, \quad \ell = 0, \ldots, L-1 \tag{1}$$

where $\mathbf{\Delta}_\ell \in \mathbb{C}^{N_{\mathrm{ray}} \times N_{\mathrm{ray}}}$ is diagonal with non-zero complex entries, $\mathbf{A}_{\mathrm{R}} \in \mathbb{C}^{N_r \times N_{\mathrm{ray}}}$ and $\mathbf{A}_{\mathrm{T}} \in \mathbb{C}^{N_t \times N_{\mathrm{ray}}}$ contain the receive and transmit array steering vectors $\mathbf{a}_{\mathrm{R}}(\phi_n)$ and $\mathbf{a}_{\mathrm{T}}(\theta_n)$ as their columns for $n = 1, \ldots, N_{\mathrm{ray}}$. N_{ray} is the number of paths. Using this, the frequency-domain channel at subcarrier k can be written in terms of different delay taps as

$$\mathbf{H}[k] = \sum_{\ell=0}^{L-1} \mathbf{H}_\ell e^{-\mathrm{j}\frac{2\pi k}{K}\ell} = \mathbf{A}_{\mathrm{R}} \mathbf{\Delta}[k] \mathbf{A}_{\mathrm{T}}^{\mathrm{H}}. \tag{2}$$

Note that the channel $\mathbf{H}_\ell$ can be approximated using the extended virtual channel model as

$$\mathbf{H}_\ell \approx \tilde{\mathbf{A}}_{\mathrm{R}} \mathbf{\Delta}_\ell^v \tilde{\mathbf{A}}_{\mathrm{T}}^{\mathrm{H}} \tag{3}$$

where $\mathbf{\Delta}_\ell^v \in \mathbb{C}^{G_r \times G_t}$ is a sparse matrix which contains the path gains of the quantized spatial frequencies in the non-zero elements. The dictionary matrices $\tilde{\mathbf{A}}_{\mathrm{T}}$ and $\tilde{\mathbf{A}}_{\mathrm{R}}$ contain the transmitter and receiver array response vectors evaluated on a grid of size G_r for the AoA and a grid of size G_t for the AoD. Due to the few scattering clusters in mmWave channels, the sparse assumption for $\mathbf{\Delta}_\ell^v$ is commonly accepted. The above approximation then leads to

$$\mathbf{H}[k] \approx \tilde{\mathbf{A}}_{\mathrm{R}} \left(\sum_{\ell=0}^{L-1} \mathbf{\Delta}_\ell^v e^{-\mathrm{j}\frac{2\pi k}{K}\ell} \right) \tilde{\mathbf{A}}_{\mathrm{T}}^{\mathrm{H}} = \tilde{\mathbf{A}}_{\mathrm{R}} \mathbf{\Delta}^v[k] \tilde{\mathbf{A}}_{\mathrm{T}}^{\mathrm{H}}. \tag{4}$$

Note that the dictionary matrices $\tilde{\mathbf{A}}_{\mathrm{R}}$ and $\tilde{\mathbf{A}}_{\mathrm{T}}$ are common to all the subcarriers due to the frequency-flat array response vectors. Hence, as shown in the main reference [1], the sparse matrices $\mathbf{\Delta}^v[k]$ for $k = 1, \ldots, K$ have the non-zero elements at the same indices. This means that they share a common sparsity pattern and this is exploited in the proposed solution.

The concatenated received signals during M training intervals for the estimation of the channel $\mathbf{H}[k]$ at the kth subcarrier are

$$\underbrace{\begin{bmatrix} \mathbf{y}^{(1)}[k] \\ \vdots \\ \mathbf{y}^{(M)}[k] \end{bmatrix}}_{\mathbf{y}[k]} = \underbrace{\begin{bmatrix} \mathbf{\Phi}^{(1)} \\ \vdots \\ \mathbf{\Phi}^{(M)} \end{bmatrix}}_{\mathbf{\Phi}} \mathbf{\Psi} \mathbf{h}^v[k] + \underbrace{\begin{bmatrix} \mathbf{n}_c^{(1)}[k] \\ \vdots \\ \mathbf{n}_c^{(M)}[k] \end{bmatrix}}_{\mathbf{n}_c[k]} \tag{5}$$

where

$$\mathbf{\Phi}^{(m)} = \left(\mathbf{q}^{(m)^{\mathrm{T}}} \mathbf{F}_{\mathrm{tr}}^{(m)^{\mathrm{T}}} \otimes \mathbf{W}_{\mathrm{tr}}^{(m)^{\mathrm{H}}} \right), \tag{6}$$

$$\mathbf{\Psi} = (\tilde{\mathbf{A}}_{\mathrm{T}}^{*} \otimes \tilde{\mathbf{A}}_{\mathrm{R}}) \tag{7}$$

and $\mathbf{h}^v[k] = \mathrm{vec}\{\mathbf{\Delta}^v[k]\} \in \mathbb{C}^{G_t G_r}$ is the sparse vector containing the complex channel gains. Note that the first matrix on the right side of the above equation, i.e, $\mathbf{\Phi} \in \mathbb{C}^{ML_r \times N_t N_r}$ is known and we are given the received signal $\mathbf{y}[k]$ for $k = 1, \ldots, K$. We will use a fixed grid although the grid points are different for training and testing stages. Hence, the dictionary matrix $\mathbf{\Psi}$ is also known.

We take the inverse FFT of the received signal sequence and scale it accordingly to keep the noise variance the same, i.e.,

$$\begin{aligned} \widetilde{\mathbf{y}}[\ell] =& \frac{1}{\sqrt{K}} \left(\sum_{k=0}^{K-1} \mathbf{y}[k] e^{\mathrm{j}\frac{2\pi k}{K}\ell} \right) \\ =& \mathbf{\Phi}\mathbf{\Psi}\widetilde{\mathbf{h}}^v[\ell] + \widetilde{\mathbf{n}}_c[\ell], \quad \ell \in \mathcal{L} \end{aligned} \tag{8}$$

where $\widetilde{\mathbf{h}}^v[\ell] = \mathrm{vec}\{\mathbf{\Delta}_\ell^v\}$ and the noise $\widetilde{\mathbf{n}}_c[\ell]$ has the same distribution as $\mathbf{n}_c[k]$. Here, $\mathcal{L} \subset \{0, \ldots, K-1\}$ denote the set of indices of the dominating delay taps. This set is determined heuristically by a simple thresholding on the total energy of the received signals $\widetilde{\mathbf{y}}[\ell]$, for $\ell = 0, \ldots, K-1$. This operation is done to increase the signal-to-noise ratio (SNR) by eliminating possibly all-noise samples.

As a next step, we apply a whitening filter. Note that the covariance matrix of $\widetilde{\mathbf{n}}_c[k]/\sigma$ is given as

$$\mathbf{C}_{\mathbf{w}} = \mathrm{blkdiag} \left\{ \mathbf{W}_{\mathrm{tr}}^{(1)^{\mathrm{H}}} \mathbf{W}_{\mathrm{tr}}^{(1)}, \ \ldots, \ \mathbf{W}_{\mathrm{tr}}^{(M)^{\mathrm{H}}} \mathbf{W}_{\mathrm{tr}}^{(M)} \right\}. \tag{9}$$

Then the whitened signal is obtained as

$$\mathbf{y}_{\mathbf{w}}[\ell] = \mathbf{C}_{\mathbf{w}}{}^{1/2}\widetilde{\mathbf{y}}[\ell] = \mathbf{C}_{\mathbf{w}}{}^{1/2}\mathbf{\Phi}\mathbf{\Psi}\widetilde{\mathbf{h}}^{v}[\ell] + \mathbf{n}_{\mathbf{w}}[\ell] \tag{10}$$

where $\mathbf{n}_{\mathbf{w}}[\ell] = \mathbf{C}_{\mathbf{w}}{}^{1/2}\widetilde{\mathbf{n}}_c[\ell] \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}_{ML_r}, \sigma^2\mathbf{I}_{ML_r})$.

II. Pattern-Coupled Sparse Bayesian Learning

In this approach, we will mainly extend the pattern-coupled sparse Bayesian learning in [2] for our problem by introducing sparsity connections between consecutive AoAs and AoDs. We will also model the hyperparameters so that they are shared by all the delay taps to exploit the common sparsity among them.

Note that the following method is first applied to the true channels from the training data by regularizing it with a very small variance white Gaussian noise and uniform grids for AoAs and AoDs. Then, in testing stage, the grid points are refined based on the joint AoA/AoD pattern that is extracted from the training data. Since the sparse model is the same except for the measurement matrices (there is an additional matrix $\mathbf{C}_{\mathbf{w}}{}^{1/2}\mathbf{\Phi}$ affecting on the true channels), we will directly present the method in testing stage, which operates on the received signals $\mathbf{y}_{\mathbf{w}}[\ell]$, for $\ell \in \mathcal{L}$.

The method in [2] assumes noisy measurement in the form of

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w} \tag{11}$$

where $\mathbf{y}$ is the observed vector, $\mathbf{A}$ is the measurement matrix, and the $\mathbf{x}$ is the sparse signal with some unknown block-sparsity patterns. The vector $\mathbf{w}$ is the zero-mean Gaussian noise with scaled identity covariance matrix. Hence, the model is in accordance with our model in (10). Let us define $\mathbf{A} = \mathbf{C}_{\mathbf{w}}{}^{1/2}\mathbf{\Phi}\mathbf{\Psi}$, $\mathbf{y}^{\ell} = \mathbf{y}_{\mathbf{w}}[\ell]$, $\mathbf{x}^{\ell} = \widetilde{\mathbf{h}}^{v}[\ell]$ and $\mathbf{w}^{\ell} = \mathbf{n}_{\mathbf{w}}[\ell]$. Then, we have all the measurements from (10) for $\ell \in \mathcal{L}$ in the form

$$\mathbf{y}^{\ell} = \mathbf{A}\mathbf{x}^{\ell} + \mathbf{w}^{\ell}, \quad \ell \in \mathcal{L}. \tag{12}$$

Let us express the sparse vector $\mathbf{x}^{\ell} \in \mathbb{C}^{G_tG_r}$ in the following form with special indices:

$$\mathbf{x}^{\ell} = \begin{bmatrix} x^{\ell}_{1,1} \\ \vdots \\ x^{\ell}_{G_r,1} \\ x^{\ell}_{1,2} \\ \vdots \\ x^{\ell}_{G_r,2} \\ \vdots \\ x^{\ell}_{1,G_t} \\ \vdots \\ x^{\ell}_{G_r,G_t} \end{bmatrix}, \quad \ell \in \mathcal{L}. \tag{13}$$

Note that the elements of $\mathbf{w}^{\ell}$ are independent and identically distributed zero-mean complex Gaussian random variables with variance σ^2.

A. Proposed Pattern-Coupled Hierarchical Model

To exploit both the block-sparse structure along AoAs, AoDs, and the common sparsity for all the delay taps, we define a prior over $\mathbf{x} \triangleq \{\mathbf{x}^{\ell} : \ell \in \mathcal{L}\}$ as

$$p(\mathbf{x}|\boldsymbol{\alpha}) = \prod_{g_r=1}^{G_r}\prod_{g_t=1}^{G_t}\prod_{\ell\in\mathcal{L}}\mathcal{N}_{\mathbb{C}}(x^{\ell}_{g_r,g_t}|0, \eta^{-1}_{g_r,g_t}) \tag{14}$$

To model the pattern-coupled block sparsity, we express the common parameter η_{g_r,g_t} among the delay taps as

$$\begin{aligned}\eta_{g_r,g_t} =& \alpha_{g_r,g_t} + \beta_r\alpha_{g_r-1,g_t} + \beta_r\alpha_{g_r+1,g_t} \\ &+ \beta_t\alpha_{g_r,g_t-1} + \beta_t\alpha_{g_r,g_t+1}\end{aligned} \tag{15}$$

where $\{\alpha_{g_r,g_t}\}$ are the hyperparameters controlling the sparsity of $\mathbf{x}$. The parameters $\beta_r \in [0,1]$ and $\beta_t \in [0,1]$ indicate the pattern relevance between $x^{\ell}_{g_r,g_t}$ and its neighboring coefficients and they are taken as known constants in accordance with the related works. Different from [2], we do not impose any Gamma prior for the hyperparameters $\{\alpha_{g_r,g_t}\}$. Instead, we do not put any prior based on our experiments.

Note that in testing stage the noise variance is not given explicitly. Instead a range information is provided. So, we assume that we don't know $\gamma = 1/\sigma^2$, but we introduce a uniform distribution prior for γ, i.e., $\gamma \sim \mathcal{U}[\gamma_{\text{low}}, \gamma_{\text{upp}}]$ where the bounds are provided along with the test data. This assumption also differs from the Gamma distribution that is considered in [2].

We will utilize an expected maximization (EM) algorithm for learning the sparse signal $\mathbf{x}$ and the hyperparameters $\Theta \triangleq \{\boldsymbol{\alpha}, \gamma\}$. In the EM formulation, the signal $\mathbf{x}$ is treated as a hidden variable, and we iteratively maximize a lower bound on the posterior probability $p(\Theta|\mathbf{y})$ (this lower bound is also referred to as the Q-function). The algorithm alternates between an E-step and a M-step.

B. E-Step

In the E-step, we need to compute the posterior distribution of $\mathbf{x}$ conditioned on the observed data and the hyperparameters estimated from the tth iteration, i.e.,

$$p(\mathbf{x}|\mathbf{y}, \Theta^{(t)}) \propto p(\mathbf{x}|\boldsymbol{\alpha}^{(t)})p(\mathbf{y}|\mathbf{x}, \gamma^{(t)}). \tag{16}$$

The posterior probability can be computed as a multivariate Gaussian distribution with mean and covariance matrix for $\mathbf{x}^{\ell}$ as

$$\boldsymbol{\mu}^{\ell(t)} = \gamma^{(t)}\left(\gamma^{(t)}\mathbf{A}^{\text{H}}\mathbf{A} + \mathbf{D}^{(t)}\right)^{-1}\mathbf{A}^{\text{H}}\mathbf{y}^{\ell}, \quad \ell \in \mathcal{L} \tag{17}$$

$$\boldsymbol{\chi}^{\ell(t)} = \left(\gamma^{(t)}\mathbf{A}^{\text{H}}\mathbf{A} + \mathbf{D}^{(t)}\right)^{-1}, \quad \ell \in \mathcal{L} \tag{18}$$

from [2] where $\mathbf{D}^{(t)} \in \mathbb{R}^{G_rG_t \times G_rG_t}$ is a diagonal matrix with the diagonal elements $\eta^{(t)}_{g_r,g_t}$ that are ordered according to the indexing in (13). Let $\mu^{\ell}_{g_r-1,g_t}{}^{(t)}$ and $\chi^{\ell}_{g_r-1,g_t}{}^{(t)}$ denote the elements of $\boldsymbol{\mu}^{\ell}$ and $\boldsymbol{\chi}^{\ell}$ corresponding to the index ordering in (13).

C. M-Step

In the M-step, the hyperparameters $\Theta = \{\boldsymbol{\alpha}, \gamma\}$ are estimated by treating $\mathbf{x}$ as hidden variables and iteratively maximizing the Q-function, i.e.,

$$\begin{aligned}\Theta^{(t+1)} =& \arg\max_{\Theta} \; Q(\Theta|\Theta^{(t)}) \\ =& \arg\max_{\Theta} \; \mathbb{E}_{\mathbf{x}|\mathbf{y},\Theta^{(t)}}\{\ln p(\Theta|\mathbf{x},\mathbf{y})\} \end{aligned} \tag{19}$$

where the expectation is with respect to the posterior distribution $p(\mathbf{x}|\mathbf{y},\Theta^{(t)})$. We can express the above maximization with respect to Θ as

$$\begin{aligned}\underset{\Theta}{\text{maximize}} \;\; & \mathbb{E}_{\mathbf{x}|\mathbf{y},\Theta^{(t)}}\{\ln p(\boldsymbol{\alpha})p(\mathbf{x}|\boldsymbol{\alpha})\} \\ & + \mathbb{E}_{\mathbf{x}|\mathbf{y},\Theta^{(t)}}\{\ln p(\mathbf{y}|\mathbf{x},\gamma)p(\gamma)\} \end{aligned} \tag{20}$$

We can implement the iterative updates in an alternating manner as follows:

1) Update for $\boldsymbol{\alpha}$: Following a similar approach in [2], we can obtain a suboptimal update for $\boldsymbol{\alpha}$ as (the optimal update is very hard due to the coupled variables)

$$\alpha_{g_r,g_t}^{\ell\ (t+1)} = \frac{|\mathcal{L}|}{\omega_{g_r,g_t}^{(t)}}, \quad \ell \in |\mathcal{L}|, \quad g_r = 1,\ldots,G_r, \quad g_t = 1,\ldots,G_t \tag{21}$$

where

$$\begin{aligned}\omega_{g_r,g_t}^{(t)} = \sum_{\ell\in\mathcal{L}} \Bigg(& \left|\mu_{g_r,g_t}^{\ell\ (t)}\right|^2 + \chi_{g_r,g_t}^{\ell\ (t)} \\ & + \beta_r \left(\left|\mu_{g_r-1,g_t}^{\ell\ (t)}\right|^2 + \chi_{g_r-1,g_t}^{\ell\ (t)}\right) \\ & + \beta_r \left(\left|\mu_{g_r+1,g_t}^{\ell\ (t)}\right|^2 + \chi_{g_r+1,g_t}^{\ell\ (t)}\right) \\ & + \beta_t \left(\left|\mu_{g_r,g_t-1}^{\ell\ (t)}\right|^2 + \chi_{g_r,g_t-1}^{\ell\ (t)}\right) \\ & + \beta_t \left(\left|\mu_{g_r,g_t+1}^{\ell\ (t)}\right|^2 + \chi_{g_r,g_t+1}^{\ell\ (t)}\right) \Bigg), \\ & g_r = 1,\ldots,G_r, \quad g_t = 1,\ldots,G_t. \end{aligned} \tag{22}$$

2) Update for γ: The hyperparameter γ, which is the inverse of the noise variance and have a uniform prior distribution on $[\gamma_{\text{low}}, \gamma_{\text{upp}}]$ can be updated by adopting the derivation in [3] to the uniform prior we consider as

$$\gamma^{(t+1)} = \arg\max_{\gamma} \; \mathbb{E}_{\mathbf{z}|\mathbf{y},\Theta^{(t)}}\left\{\ln p(\gamma)p(\mathbf{y}|\mathbf{z},\gamma)\right\}. \tag{23}$$

Using the uniform prior, we can obtain $\gamma^{(t+1)}$ as

$$\gamma^{(t+1)} = \Pi_{\gamma}\left(\frac{ML_r|\mathcal{L}|}{\sum_{\ell\in\mathcal{L}}\left(\|\mathbf{y}^{\ell} - \mathbf{A}\boldsymbol{\mu}^{\ell}\|^2 + \left(\gamma^{(t)}\right)^{-1}\left(G_rG_t - \text{tr}\left(\boldsymbol{\chi}^{\ell\ (t)}\mathbf{D}^{(t)}\right)\right)\right)} \right) \tag{24}$$

where

$$\Pi_{\gamma}(x) = \begin{cases} \gamma_{\text{low}} & \text{if } x \le \gamma_{\text{low}} \\ x & \text{if } \gamma_{\text{low}} < x \le \gamma_{\text{upp}} \\ \gamma_{\text{upp}} & \text{if } x > \gamma_{\text{upp}} \end{cases}. \tag{25}$$

The overall EM algorithm is implemented by applying the updates iteratively until the difference between $\boldsymbol{\mu}^{\ell\ (t)}$ and $\boldsymbol{\mu}^{\ell\ (t-1)}$ is negligibly small. At the final iteration, the sparse vector estimate $\widehat{\mathbf{x}}^{\ell}$ is set to $\boldsymbol{\mu}^{\ell\ (t)}$, for $\ell \in \mathcal{L}$ and after multiplying it with the dictionary matrix $\boldsymbol{\Psi}$, we obtain the time-domain channel estimates at the dominant delay taps in $\mathcal{L}$. Then, we take the K-point FFT of the time channels and scale by $1/\sqrt{K}$ to obtain the final frequency channel estimates.

Now, we will describe the overall method in the next section in more detail.

III. The Selected Method for Learning the Joint Relations Between AoAs and AoDs

As a first step, we construct the dictionary matrix $\boldsymbol{\Psi}$ by $G_r = 96$ AoA and $G_t = 24$ AoD grid points that are uniformly selected from $[0, \pi]$. We only consider this angle range since the array steering vectors for the other angles are the same as those with the angles in $[0, \pi]$. Then using 10 000 true frequency channels provided in the training data set, we add a white Gaussian complex noise to the time-domain channels to obtain the sparse model

$$\mathbf{y}_{\text{training}}^{\ell} = \boldsymbol{\Psi}\mathbf{x}_{\text{training}}^{\ell} + \mathbf{w}_{\text{training}}^{\ell}. \tag{26}$$

Note that the variance of the noise is selected as very small value, i.e., 10^{-4}. The motivation is to regularize the model and apply the EM algorithm described previously without any issue. We apply the EM algorithm in the previous section by keeping the inverse noise variance $\gamma = 10^4$ fixed to all the 10 000 models obtained by training dataset. Then using all the sparse estimates $\widehat{\mathbf{x}}_{\text{training}}^{\ell}$, we estimate the power distribution along $2G_r = 192$ AoA points and $2G_t = 48$ AoD points as in Fig. 1. Here, we apply an interpolation to both AoA and AoD axis since we will utilize this in the grid construction algorithm in testing stage. As Fig. 1 shows, some AoA/AoD grid points are more probable for the given simulation site. To exploit this learned information, we propose a grid construction algorithm to locate the grid points more densely in the yellow regions compared to blue regions. We first start with a uniform grid for both AoA and AoD in $[0, \pi)$ with $96 \cdot 24$ points in total. Then, we assign additional $96 \cdot 8$ grid points to the most yellowish regions in Fig. 1 by sorting the power values accordingly. As a next stage, we change the locations of the points to move them to the places where the power of the sparse vectors obtained from the training data is greater. At the same time, we try to prevent the neighboring grid points from being far away by some tuning and adjustments. At the end, the constructed grid point map is shown in Fig. 2 where the yellow points denote the selected $96 \cdot 32$ grid points to be utilized in constructing the dictionary matrix in the testing stage.

In the testing stage, after constructing the dictionary matrix $\boldsymbol{\Psi}$ according to the pattern in Fig. 2, we also modify the pattern-coupling relations accordingly. For this new grid

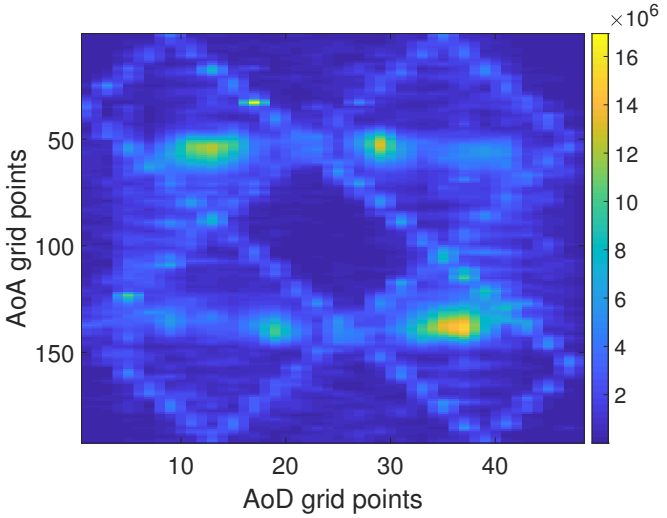

Fig. 1. Heatmap for the power distribution of the sparse vector among AoA and AoD grid points.

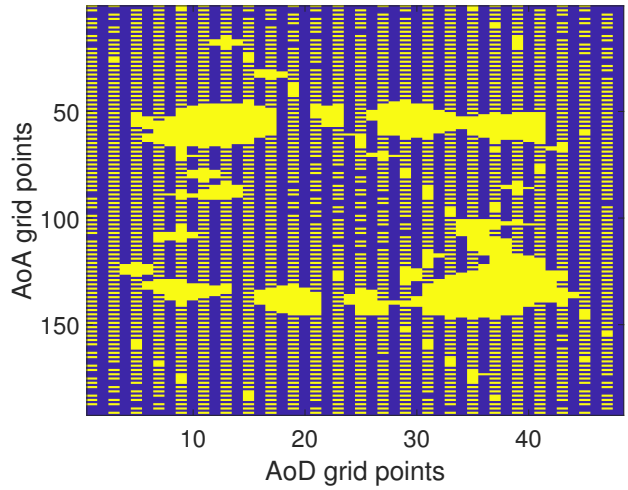

Fig. 2. Non-uniform grid pattern for AoA and AoD in testing stage of the algorithm. The yellow pixels correspond to the selected $96 \cdot 32$ grid points.

structure, the AoA and AoD pattern-coupled block sparsity relations in (15) and (22) are modified such that the consecutive AoA and AoD gird points in Fig. 2 are constructed as coupled by keeping only the pairs with some distance threshold. The updates in the EM algorithm are the same except for the indices according to the pattern-coupled block sparsity pattern.

IV. Summary

We have adopted an EM-based sparse Bayesian learning method for the competition problem to exploit the shared sparsity between different delay taps and possible sparsity pattern couplings between consecutive AoAs and AoDs. We have applied the algorithm to the time-domain received signals by only keeping the most dominant delay taps to increase SNR. First, we have applied the pattern-coupled Sparse Bayesian learning algorithm to the true channels in the training dataset by adding a small noise to regularize them. Using the considered method, we have obtained all the sparse representations for the channels in the provided dataset. Then, using the respective sparse vectors and exploiting the density map of joint AoA/AoD grids, we have selected a non-uniform grid and refined the pattern couplings between hyperparameters. The algorithm is applied to the test dataset to obtain the channel estimates. Note that only the true frequency channels in the training data, the received signals in the test data, and the SNR intervals are utilized in the considered framework.

References

[1] J. Rodríguez-Fernandez, N. González-Prelcic, K. Venugopal, and R. W. Heath, "Frequency-domain compressive channel estimation for frequency-selective hybrid millimeter wave MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 5, pp. 2946–2960, 2018.

[2] J. Fang, L. Zhang, and H. Li, "Two-dimensional pattern-coupled sparse bayesian learning via generalized approximate message passing," *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2920–2930, 2016.

[3] J. Fang, Y. Shen, H. Li, and P. Wang, "Pattern-coupled sparse bayesian learning for recovery of block-sparse signals," *IEEE Transactions on Signal Processing*, vol. 63, no. 2, pp. 360–372, 2015.