Time-Varying Massive MIMO Channel Estimation: Capturing, Reconstruction, and Restoration

Muye Li, Shun Zhang, Member, IEEE, Nan Zhao, Senior Member, IEEE, Weile Zhang, Member, IEEE, and Xianbin Wang, Fellow, IEEE

Abstract—To estimate time-varying MIMO channel at base station, traditional downlink (DL) channel restoration schemes usually require the reconstruction for the covariance of downlink process noise vector, which is dependent on DL channel covariance matrix (CCM). However, the acquisition of the CCM leads to extremely high overhead in massive MIMO systems. To tackle this problem, we propose a novel scheme for DL channel tracking in this paper. First, by utilizing virtual channel representation (VCR), we develop a dynamic uplink (UL) massive MIMO channel model with the consideration of off-grid refinement. Then, a coordinate-wise expectation maximization (EM) algorithm is adopted for capturing model parameters, including the spatial signatures, time-correlation factors, off-grid bias, channel power, and noise power. By exploiting the UL/DL angle reciprocity, the spatial signatures, time-correlation factors and off-grid bias of the DL channel model can be reconstructed with the knowledge of UL. However, channel power and noise power are closely related with the carrier frequency, which cannot be perfectly inferred from the UL. Instead of discovering these two parameters with dedicated training, we resort to the optimal Bayesian Kalman filter (OBKF) method to accurately track the DL channel with partial prior knowledge. At the same time, the model parameters will be gradually restored. Specially, the factor-graph and the Metropolis Hastings MCMC are utilized within the OBKF framework. Finally, numerical results are provided to demonstrate the efficiency of our proposed scheme.

Index Terms—Massive MIMO, off-grid, sparse Bayesian learning, optimal Bayesian Kalman filter, factor graph.

I. INTRODUCTION

A S MASSIVE multiple-input multiple-output (MIMO) tremendously improved spectral and energy efficiency [1], it has become the most critical technology for the 5th generation (5G) and beyond wireless networks to support the dramatically growing data traffic [2]–[5]. In order to exploit the advantages of massive MIMO, it is indispensable for the base station (BS) to obtain precise channel state information (CSI). However, in the frequency-division duplex (FDD) systems, obtaining CSI in the same way will incur significant channel estimation overhead together with increased pilot contamination [6]–[9].

To reduce the overhead, a set of new strategies were proposed to reduce the effective channel dimensions, where low-rank characteristics of the covariance matrix for massive MIMO channel was fully exploited [10]–[15]. In [10], a joint spatial division multiplexing (JSDM) scheme was proposed, which required reduced channel estimation downlink training overhead and CSI feedback. Nam et al. extended this approach, and proposed a simple scheme for user grouping and a low-overhead probabilistic scheduling algorithm [11]. In [12], Adhikary et al. modified the JSDM scheme to reduce its computational complexity. Under the two-stage precoding framework, Sun et al. proposed the beam division multiple access transmission scheme for FDD massive MIMO system in [13], where the downlink only required the statistical CSI. Xie et al. proposed a new channel estimation scheme for TDD/FDD massive MIMO system [14], where the UL/DL CCMs were reconstructed without any additional training cost. In [15], Han et al. proposed an DL channel reconstruction scheme, where the feedback was limited. The aforementioned methods in common used the spatial property for the implementation of orthogonal transmission to a number of users. Theoretically, the spatial property can be derived from CCM. Thus, the low complexity and efficiency methods for obtaining CCM are significant to the above studies [10]–[15].

Nevertheless, it is quite difficult to acquire channel covariances in massive MIMO system, due to the complexity of singular value decomposition (SVD) for high-dimensional matrix [16]. To overcome this challenge, Zeng et al. proposed a PDM method for MIMO spatial multiplexing with very low complexity, which was applicable for both narrow-band and wide-band communications [17]. Xie et al. developed a spatial basis expansion model (SBEM) with the aid of antenna array theory, and could carry out UL and DL channel...
estimations of multiusers without channel covariances [18].
Moreover, there are some interesting works about the time-
varying channel estimation over the massive MIMO networks.
Dahiya et al. examined the channel tracking for block-fading
channels in massive MIMO systems [19]. However, it did not
fully exploit the spatial sparse characteristics. Furthermore,
in [20], Qin et al. proposed one effective time-varying channel
estimation scheme for the massive MIMO-OFDM system,
where the complex exponential BEM (CE-BEM) was utilized
to depict the time-varying channels. The authors in [20]
focused on the downlink, and resorted to one compressive
sensing (CS) algorithm to achieve the channel information.
However, [20] only utilized the CE-BEM to depict the time-
varying characteristics of the massive channels within one
OFDM symbol, and did not consider the time-correlation
between adjacent symbols. Our previous work [21] proposed
a scheme for time-varying massive MIMO channel tracking.
The authors developed an EM-based sparse Bayesian learning
(SBL) framework to learn the temporal correlation factor,
and the channel powers, where Kalman
filter (KF) and Rauch-Tung-Striebel smoother (RTSS) were
adopted. Then we applied a KF with reduced dimension for
the channel tracking. However, the channel powers are closely
related with the carrier frequency, which can not be perfectly
infereed from the UL ones. Moreover, in [21], we did not
incorporate the noise covariance into the model parameters.
Finally, considering the randomness of direction of arrivals
(ROAs) of impinging signals, as spatial sample mismatching
and power leakage exists, it is unavoidable to lose performance
with the channel tracking schemes in [20], [21].

Motivated by aforementioned works and existing problems,
this paper focuses on the DL channel restoration for the time-
varying massive MIMO channel. We will directly capture
the parameters in the UL channel model instead of requiring
and analyzing the CCMs. First, a time-varying off-grid massive
MIMO channel model with the adoption of Taylor series
and the VCR [22] is constructed, Then, a SBL framework is
derived to estimate the parameters of the sparse virtual channel
model and the noise covariance. To avoid the unacceptable
complexity, we apply a coordinate-wise EM algorithm. Next,
according to the spatial signatures, we use a unified low
dimensional KF for the UL tracking. Thanks to the UL/DL
angle reciprocity, the time-correlation factors, the DL spatial
signatures and the off-grid bias can be directly obtained from
UL containers. Unfortunately, the other two model parameters
can not be perfectly inferred from the UL’s characteristics,
as they are closely related with the carrier frequency. Although
we can still use the method for the UL to capture the DL model
parameters, this would inevitably introduce a tremendous scale
of overheads. In order to avoid this obstacle, we resort to
the OBKF method to accurately track the DL channel with
the partially prior knowledge. We first show the recursive
equations of the DL channel restoration. Then, we employ
an MCMC method to approximate the two posterior noise
statistics. Finally, to obtain the posterior distribution of the
noise second-order statistics, a factor graph based sum-product
algorithm is introduced.

The rest of this paper is organized as follows. Section II
gives the system model and the description of virtual channel
model. The main ideas of the Coordinate-wise EM algorithm
for UL model parameters learning and a concise depiction of
UL channel tracking is illustrated in Section III. Section IV
presented the DL model parameters recovering and the DL
virtual channel tracking by the proposed OBKF method. The
simulation results are given in Section V, and Section VI shows
the conclusions.

\textbf{Notations:} Denote lowercase (uppercase) boldface as vector
(matrix). \((\cdot)^{H}\) and \((\cdot)^{T}\) represent the Hermitian and the trans-
pose matrix, respectively. \(I_{N}\) represents a \(N \times N\) identity
matrix. \(E\{\cdot\}\) is the expectation operator. Denote \(\text{tr}\{\cdot\}\) and \(|\cdot|\)
as the trace and the determinant of a matrix, respectively.
We use \([A]_{i,j}\) and \(A_{Q}:(\text{or } A_{Q})\) to represent the \((i, j)\)-th
entry of \(A\) and the submatrix of \(A\) which contains the rows
(or columns) with the index set \(Q\), respectively. \(x_{Q}\) is the
subvector of \(x\) built by the index set \(Q\). \(v \sim \mathcal{C}\mathcal{N}(0, I_{N})\) means
that \(v\) satisfies the complex Gaussian distribution with zero
mean and covariance \(I_{N}\). \([p]\) denotes the largest integer less
than or equal to \(p\). \(E\{\cdot|\cdot\}\) denotes the set \(E\{\cdot|\cdot\}\)
element \(\alpha^{(l-1)}\). The real component of \(x\) is expressed as
\(\Re(x)\). \(\text{diag}(x)\) is a diagonal matrix whose diagonal elements
are formed the elements of \(x\), while \(\text{blkdiag}(X_{1}, X_{2}, \ldots)\) is
a block diagonal matrix formed by \(X_{1}, X_{2}, \ldots\).

\section{System Model and Channel Characteristics}

Consider an uplink massive MIMO network, where \(N_{t} \gg 1\)
uniform linear array (ULA) antennas are equipped at the BS,
and \(K\) users with single-antenna are randomly distributed
in the area. We adopt a geometric channel model with \(L\)
emerging paths from the \(k\)-th user. Denote \(\theta_{k,l,m}\) as a DOA
of \(k\)-th user, \(l\)-th path and \(m\)-th time block, and the BS antenna
array spatial steering vector can be defined as:

\[ a(\theta_{k,l,m}) = \left[ 1, e^{j2\pi\frac{d}{\lambda}\sin(\theta_{k,l,m})}, \ldots, e^{j(N_{t}-1)2\pi\frac{d}{\lambda}\sin(\theta_{k,l,m})} \right]^{T}, \]

where \(d \leq \lambda/2\) is antenna spacing of the BS; \(\lambda\) is the carrier
wavelength.

It is assumed that the DOA of each path is quasi-static
during a block of \(L_{c}\) and changes from block to block.
The system sampling rate is \(\frac{1}{T_{c}}\). Then, the uplink channel
\(h_{k,m} \in \mathbb{C}^{M \times 1}\) from the \(k\)-th user to the BS during the \(m\)-th
block can be written as [25], [26]

\[ h_{k,m} = \int_{-\infty}^{+\infty} \sum_{l=1}^{L} a(\theta_{k,l,m}) e^{j2\pi\nu_{L_{c}} T_{c}} h_{k}(\theta_{k,l,m}, \nu) d\nu, \]

where \(h_{k}(\theta_{k,l,m}, \nu)\) denotes the joint angle-Doppler channel
gain function of the \(k\)-th user.

As in [27], the VCR can be utilized to dig the sparsity of
\(h_{k,m}\) as \(h_{k,m} = F_{N_{t}} h_{k,m}\), where \(h_{k,m}\) is the virtual channel
of \(h_{k,m}\), and \(F_{N_{t}}\) is the \(N_{t} \times N_{t}\) unified discrete Fourier
transformation (DFT) matrix with \((p, q)\)-th entry as \(F_{N_{t}}[p, q] = \frac{1}{\sqrt{N_{t}}} e^{-j\frac{2\pi pq}{N_{t}}})\). Furthermore, we can adopt the sparse signal
model to depict the dynamics of $\hat{h}_{k,m}$ by adopting the first order auto regressive (AR) model [29] as
\[
\begin{align*}
\hat{h}_{k,m} &= \text{diag}(c_k) r_{k,m}, \\
r_{k,m} &= \alpha_k r_{k,m-1} + \upsilon_{k,m},
\end{align*}
\]
(3)
where the time-varying variable $r_{k,m}$ is a Gaussian Markov vector, $\alpha_k$ is the transmission factor, $\upsilon_{k,m} \sim \mathcal{CN}(0, \Lambda_k)$ is the process noise vector where $\Lambda_k = \text{diag} \left( \{ \lambda_{k,1}^{2}, \lambda_{k,2}^{2}, \ldots, \lambda_{k,N}^{2} \} \right)$, and the spatial signature [30] vector $c_k$ is determined by the set
\[
\mathcal{Q}_k = \left\{ p \mid N_1 \frac{d}{\lambda} \sin(\theta_{k,m}^{\text{min}}) \leq p \leq N_1 \frac{d}{\lambda} \sin(\theta_{k,m}^{\text{max}}), p \in \mathbb{Z} \right\},
\]
(4)
as $|c_k|_p = 1$ when $p \in \mathcal{Q}_k$, where $n$ denotes the index of the dominant paths of DOAs.

Then we take the off-grid model into consideration. In fact, the DFT basis can be seen as a predefined spatial sampling grid for the impinging signals, and discretely covers the entire angle domain. However, in real transition processes, the DOAs would not exactly imping on those grids, and the direction mismatching emerges. Under such circumstance, we define the bias vector $\rho_k$, and derive a bias added DFT matrix, whose spatial index will be added with $\rho_k$, i.e. $p' = p + [\rho_k]_p$. Correspondingly, the set $\mathcal{Q}_k$ can be redefined as:
\[
\mathcal{Q}_k = \left\{ p \mid p + \rho_p = N_1 \frac{d}{\lambda} \sin(\theta_{k,m}^{\text{min}}), p \in \mathbb{Z} \right\},
\]
(5)
where $\rho_{k,l} \in [-0.5, 0.5]$.

An example is illustrated in Fig. 1. It intuitively explains the relationship between the spatial parameters and the virtual channel vector.

For simplicity, we use $\mathbf{A}$ to represent $\mathbf{F}_{N_t}$. Then, the channel vector $\mathbf{h}_{k,m}$ can be derived with the Taylor series expansion as
\[
\begin{align*}
\mathbf{h}_{k,m} &= \mathbf{A}^{H} + \mathbf{B}^{H} \text{diag}(\rho_k) \mathbb{1}_{\mathcal{Q}_k} [\hat{\mathbf{h}}_{k,m}]_{\mathcal{Q}_k} \\
&= \left[ \Phi(\rho_k) \right]^{H} [\hat{\mathbf{h}}_{k,m}]_{\mathcal{Q}_k},
\end{align*}
\]
(6)
where $[\mathbf{B}^{H}]_{p}$ is obtained through taking derivative of $[\mathbf{A}^{H}]_{p}$ with respect to $p$; every element of $\rho_k$ is the bias added on the corresponding predefined grid.

**Remark 1**: If the spatial signature $c_k$ and the basis vector $\rho_k$ can be accurately estimated, the DOAs can match the bias-added DFT matrix, and the power leakage can be weak but not be entirely removed. One effective way to solve that is to directly estimate the real DOAs rather than the position of sparse points, which have been proposed in [27].

However, the advantage of the DFT-based description is the low complexity in the massive MIMO channel estimation. Furthermore, the massive MIMO estimation results on basis of the DFT-based description can provide one good initial point, which can decrease the complexity of the real DOA recovering.

We can rewrite the AR model of the practical channel as
\[
\begin{align*}
\mathbf{h}_{k,m} &= \Phi(\rho_k)^{H} \text{diag}(c_k) \mathbf{r}_{k,m}, \\
\mathbf{r}_{k,m} &= \alpha_k \mathbf{r}_{k,m-1} + \mathbf{v}_{k,m},
\end{align*}
\]
(7)

Notice that, in (7), $c_k$ and $\rho_k$ are the spatial signatures and the DOA bias of the $k$-th user, while both $\Lambda_k$ and $\alpha_k$ are the parameters of the virtual channel. As the AR model constructed, the learning of the channel statistical parameters is equivalent to capturing the model parameters $\rho_k$, $\alpha_k$, $c_k$, $\Lambda_k$. Moreover, if the users move at not too high speed, the model parameters can be treated as constant during many coherence blocks [21].

### III. Model Parameters Capturing via Uplink Training and Uplink Channel Tracking

Before proceeding, we give the the overall block diagram Fig. 2 to coarsely introduce our work. Firstly, during several dedicated training blocks, we capture the model parameters for the UL with EM algorithm. Secondly, we track the UL channels with the classical KF, and implement the data transmission. Thirdly, with the angle reciprocity, we reconstruct some parameters for the DL channels. Fourthly, in some time blocks, we adopt OBKF method to accurately track the DL channels with partial model knowledge. With the above operation, we can gradually restore the DL model parameters, which can not be perfectly inferred from the UL ones. Notice that the DL data transmission is implemented at the same time. Fifthly, we do the similar operation with the second step for the DL data transmission. Finally, we find that the model parameters change drastically, and then restart the parameter capturing process.

Without loss of generality, we assume that $\tau \leq K$ orthogonal training sequences of length $L_s \leq L_c$ is allocated in the current cell. Denote $\mathbf{S} = [s_1, s_2, \ldots, s_T]$ with $s_i^{H} s_j = L_c \sigma_n^2 \delta(i-j)$ as the orthogonal training set, where $\sigma_n^2$ is the pilot power. We assume that $K = C\tau$, where $C$ is the number of user group for illustration simplicity.

We take one group as an example, and $M$ channel blocks are used to learn the channel model parameters. The received signal during the $m$-th block can be written as
\[
\mathbf{Y}_m = \sum_{k=1}^{\tau} \mathbf{h}_{k,m} \mathbf{s}_k^T + \mathbf{N}_m
\]
\[
= \sum_{k=1}^{\tau} \Phi(\rho_k)^{H} \text{diag}(c_k) \mathbf{r}_{k,m} \mathbf{s}_k^T + \mathbf{N}_m,
\]
(8)
where $\mathbf{N}_m$ denotes the independent additive complex Gaussian noise matrix with each elements subject to i.i.d. $\mathcal{CN}(0, \sigma_n^2)$, and $\sigma_n^2$ is unknown. Moreover, we define the $N_i L_s \times 1$ vector $\mathbf{y}_m = \text{vec} (\mathbf{Y}_m)$ and $N_i L_s \times 1$ vector $\mathbf{n}_m = \text{vec} (\mathbf{N}_m)$.
Then (8) can be rearranged as
\[ y_m = \sum_{k=1}^{\tau} (s_k \otimes \Phi_r(\rho_k))^H \text{diag}(c_k) r_{k,m} + n_m = J r_m + n_m, \]
(9)
where \( n_m \sim \mathcal{CN}(0, \sigma_n^2 I_{N_l L_s}) \), \( J = [J_1, J_2, \ldots, J_r] \in \mathbb{C}^{N_l L_s \times N_r} \), \( r_m = [r_{1,m}, r_{2,m}, \ldots, r_{r,m}]^T \in \mathbb{C}^{N_l L_s \times 1} \). For further use, we define the \( N_l L_s M \times 1 \) vector \( y = [y_1^T, y_2^T, \ldots, y_M^T] \), the \( N \tau M \times 1 \) vector \( r = [r_1^T, r_2^T, \ldots, r_{\tau N}^T] \), the \( N \times 1 \) vector \( \alpha = [\alpha_1, \alpha_2, \ldots, \alpha_{\tau N}]^T \), the \( N \times 1 \) vector \( c = [c_1^T, c_2^T, \ldots, c_{\tau N}^T] \), and the \( \tau N \times \tau N \) matrix \( \Lambda = \text{blkdiag}(\Lambda_1, \Lambda_2, \ldots, \Lambda_r) \).

Here, the task of the preamble is to capture the parameter set \( \Xi = \{\rho, c, \alpha, \Lambda, \sigma_n^2\} \) with the observation model (9) and the state equation (7).

A. Problem Formulation

The target of the capturing is to reach the optimal parameters set \( \hat{\Xi} \) with the observation vector \( y \). Theoretically, the classical EM ML estimator for \( \hat{\Xi} \) can be expressed as
\[ \hat{\Xi} = \arg \max_{1 \leq k \leq 0, \lambda_k, p \geq 0, ||p_k|| < 0.5} \ln p(y; \Xi). \]
(10)

Obviously, due to its high dimensional, such estimator is unadaptable to directly achieve the optimal solution. However, there is an iterative access to reach the solution via the EM algorithm [31]. Furthermore, in order to achieve a faster convergence with a correspondingly lower complexity, we will perform the coordinate-wise EM algorithm in the following.

B. Coordinate-Wise EM to Accomplish Simultaneously Sparse Signal Learning

Similar to classical EM, the coordinate-wise EM produces a sequence of \( \Xi^{(l)} \), \( l = 1, 2, \ldots \) successively, but individually deals with each parameter in \( \Xi \). Within the expectation step, this method updates only one parameter at a time, letting the other parameters unchanged [32].

To simplify the expression, we take the parameter \( \alpha \) as an example to show the process during each iteration:

- **Expectation step (E-step)**

  \[ Q(\alpha, \hat{\Xi}^{(l-1)}) = \mathbb{E}_{r[y; \hat{\Xi}^{(l-1)}]} \{ \ln p(y, r; \alpha, \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}) \}. \]

(11)

In the \( l \)-th iteration, the aim of E-step is to update those objective functions, while the M-step aims to update the estimation \( \hat{\Xi}^{(l)} \) by maximizing the current expectation function [33].

C. Expectation Step

In this subsection, we will carefully derive the objective functions. Now, we first examine \( Q(\alpha, \hat{\Xi}^{(l-1)}) \).

As the observed sample \( y \) is known, the objective function \( Q(\alpha, \hat{\Xi}^{(l-1)}) \) can be expressed as
\[ Q(\alpha, \hat{\Xi}^{(l-1)}) = \mathbb{E}_{r[y; \hat{\Xi}^{(l-1)}]} \{ \ln p(y | r; \alpha, \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}) \} + \mathbb{E}_{r[y; \hat{\Xi}^{(l-1)}]} \{ \ln p(r; \alpha, \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}) \}. \]
(13)

From (9), we obtain the conditional PDF as:
\[ p(y_m | r_m; \alpha, \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}) \sim \mathcal{CN}(\sum_{k=1}^{\tau} (J_k r_{k,m}), \sigma_n^2 I_{N_l L_s}). \]
(14)

Meanwhile, we have
\[ \ln p(r; \alpha, \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}) = \sum_{k=1}^{\tau} \ln p(r_{k,1}) + \sum_{m=2}^{M} \sum_{k=1}^{\tau} \ln p(r_{k,m} | r_{k,m-1}; \alpha, \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}), \]
(15)

where the conditional PDF \( p(r_{k,m} | r_{k,m-1}; \alpha, \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}) \) can be written as
\[ p(r_{k,m} | r_{k,m-1}; \alpha, \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}) \sim \mathcal{CN}(r_{k,m}; c_k r_{k,m-1}; \alpha^{(l-1)}) | c_k r_{k,m-1} | \hat{\Xi}^{(l-1)} | \alpha^{(l-1)}). \]
(16)

Before proceeding, we define three posterior statistics about \( r_m \), i.e.,
\[ \Pi_{k,m}^{(l)} \triangleq \mathbb{E} \{ r_{k,m} | \hat{\Xi}^{(l-1)} \}, \Theta_{k,m}^{(l)} \triangleq \mathbb{E} \{ r_{k,m}^H | \hat{\Xi}^{(l-1)} \}, \text{and } t_{k,m}^{(l)} \triangleq \mathbb{E} \{ r_{k,m}^H r_{k,m} | \hat{\Xi}^{(l-1)} \}. \]
Then, plugging (14)-(16) into (13) and taking
some reorganizations, we can obtain $Q(\alpha_k, \hat{U}^{(l-1)})$ as (17), on the top of this page where $C_1$ is the sum of items not related with $\alpha_k$.

By doing similar process of (14)-(17), we can derive other objective functions as (18)-(21), on the top of this page, where $\tilde{\Lambda}^{(l-1)}_k = (s_k \otimes \Phi(\hat{\rho}^{(l-1)}_k)^H) \text{diag}(\hat{c}^{(l-1)}_k)$, $\Psi(\hat{\rho}_k, c_k) = \text{diag}(c_k) \Phi(\hat{\rho}_k) \Phi(c_k)^H \text{diag}(c_k)$, and $C_2$, $C_3$, $C_4$, $C_5$ are not related to their own objective parameter.

From (17)-(21), it can be found that those expectation functions are dependent on $r^{(l-1)}_{k,m}, \Theta^{(l-1)}_{k,m}$, and $\Pi^{(l-1)}_{k,m-1,m}$. Similar to [21], with $\gamma$ and $\Xi^{(l-1)}$, acknowledged the above three terms can be achieved from the space-state model as

$$r_m = \hat{X}^{(l-1)} r_{m-1} + u_m, \quad y_m = \hat{J}^{(l-1)} r_m + n_m, (22)$$

where $u_m = [u^T_{1,m}, u^T_{2,m}, \ldots, u^T_{M,m}] \sim \mathcal{CN}(0, \hat{\sigma}^{2(l-1)} \otimes I_{N,r})$, $n_m \sim \mathcal{CN}(0, \hat{\sigma}^{2(l-1)} \otimes I_{N,r})$, and

$$\hat{X}^{(l-1)} = \text{diag}(\hat{\alpha}^{(l-1)}_1, \hat{\alpha}^{(l-1)}_2, \ldots, \hat{\alpha}^{(l-1)}_r) \otimes I_{N,r}, \quad \hat{\Lambda}^{(l-1)} = \text{blkdiag}(\hat{\Lambda}^{(l-1)}_1, \hat{\Lambda}^{(l-1)}_2, \ldots, \hat{\Lambda}^{(l-1)}_r), \quad \hat{\gamma}^{(l-1)} = \text{blkdiag}(\hat{\gamma}^{(l-1)}_1, \hat{\gamma}^{(l-1)}_2, \ldots, \hat{\gamma}^{(l-1)}_r).$$

D. Maximization Step

In this step, we will obtain $\Xi^{(l)}$ through maximizing all the objective function, respectively. As shown in (17)-(21), $\Xi_k$ of different users are not related, so the parameters for dynamic virtual channel of each user can be studied independently. Therefore, we will solve the maximal problem above one by one and solve them for each users independently.

1) Searching $\hat{c}^{(l)}$: It can be checked that $[A_{k,j}]_{j,j}$ is nearly 0 when $j \notin Q_k$. Based on this observation, we use a novel searching algorithm to reach a solution for $\hat{c}^{(l)}_k$.

We can obtain that $\tilde{\Lambda}^{(l)}_k$ only has a few continuous non-zero elements at its diagonal, while its other diagonal elements are nearly zero. Fig. 3(a) shows the sketch for diagonal elements of $\tilde{\Lambda}^{(l)}_k$. If we obtain the position of those non-zero point, we will find the optimal solution. An easy alternative method is to obtain the position of a big increment and the position of a big decrement through forward search.

As there exists estimation error in $\hat{\Lambda}_k$, small perturbations would happen at each diagonal element of $\hat{\Lambda}_k$. To solve this problem, we adopt the filtering method to smooth the perturbations, as shown in algorithm 1. First, we set all entries of $c_k$ to zero. Denote $s1 = [\hat{\Lambda}^{(l-1)}_1]_{j,j} + [\hat{\Lambda}^{(l)}_k]_{j,j+1} + [\hat{\Lambda}^{(l)}_k]_{j,j+2}$ and $s2 = [\hat{\Lambda}^{(l)}_k]_{j,j+3} + [\hat{\Lambda}^{(l)}_k]_{j,j+4} + [\hat{\Lambda}^{(l)}_k]_{j,j+5}$, then, we compare the two value. Let us denote $|d|_y = \ln(s^2)$ as a logarithmic function for $s^2$ and track it. We can find several couple of local maximum point $p_{ot}$ and local minimum point $p_{en}$. For each pair of the above two points, we set all the elements between $[c_k]_{p_{ot}}$ and $[c_k]_{p_{en}}$ as 1, until the power efficiency reaches an acceptable rate. Fig. 3(b) shows the position searching part of algorithm 1.
2) Computing $\hat{c}_k^{(l)}$: Taking the derivatives of (20) with respect to $[\hat{\rho}_k]^j$, we have

$$\frac{\partial Q(\hat{\rho}_k, \hat{c}_k^{(l-1)})}{\partial [\hat{\rho}_k]^j} = \frac{2}{\sigma_n^2} \sum_{m=1}^{M} \Re \left\{ \text{diag}(\hat{c}_k^{(l-1)}) \text{diag}(\hat{r}_{k,m})^{H} (s_k^H \otimes B)y_{m,j} \right\}$$

$$- \frac{2}{\sigma_n^2} \sum_{m=1}^{M} \Re \left\{ [B^{H} A \otimes \Theta_{k,m}, j,j] \hat{c}_k^{(l-1)} \right\}$$

$$- \frac{2}{\sigma_n^2} \sum_{m=1}^{M} \left\{ \text{diag}(\hat{c}_k^{(l-1)}) B^{H} B \text{diag}(\hat{c}_k^{(l-1)}) \right\} \Theta_{k,m}, j,j \times [\hat{\rho}_k]^j.$$  

Then, $[\hat{\rho}_k^{(l)}]^j$ can be derived by setting (27) to zero, i.e.,

$$\frac{\partial Q(\hat{\rho}_k, \hat{c}_k^{(l-1)})}{\partial [\hat{\rho}_k]^j} = 0,$$

and the rough solution $[\hat{\rho}_k^{(l)}]^j$ can be computed as (28), on the top of the next page.

With the constraint that $[\hat{\rho}_k]^j \in [-\frac{1}{2}, \frac{1}{2}]$, so if $[\hat{\rho}_k^{(l)}]^j \geq \frac{1}{2}$ or $[\hat{\rho}_k^{(l)}]^j \leq -\frac{1}{2}$, the result of $\hat{\rho}_k^{(l)}$ should be bounded as $\frac{1}{2}$ and $-\frac{1}{2}$, respectively.

3) Computing $\hat{\alpha}_k^{(l)}$, $\hat{\Lambda}_k^{(l)}$, and $(\hat{\sigma}_n^{(l)})^2$: The computation of these three parameters is much harder than the above one. After some calculations, we can obtain $\hat{\alpha}_k^{(l)}$, $\hat{\Lambda}_k^{(l)}$, and $(\hat{\sigma}_n^{(l)})^2$ as (29)–(31), on the top of the next page.

### E. UL Virtual Channel Tracking

Once the parameters of the virtual channel model $\Xi_k = \{\alpha_k, \Lambda_k, c_k, \rho_k, \sigma_n^2\}$ have been captured in the learning phase, to eliminate the pilot contamination and realize the simultaneous training of different users with less orthogonal training sequences, the users with non-overlapping spatial signatures are allocated to the same group, i.e.,

$$c_k c_k^T = 0.$$  

(32)

Assume that the users are divided into $G$ groups utilizing (32). Since the users in the same group can be distinguished by different spatial signatures, same training sequences for each user in one group can be allocated to estimate the virtual channel $\hat{h}_{k,m}$. Meanwhile, different user groups will be allocated with orthogonal training sequences. Therefore, we build a $G \times G$ matrix $S_G$ with $S_G^{H} S_G = G \sigma_p^2 I_G$. Then, group $g$ will be given $s_g = [s_G]_{:,g}$, and all $K$ users simultaneously transmit their sequences. Thus, the observed signal can be expressed as

$$Y_m = \sum_{g=1}^{G} \sum_{k \in G_g} h_{k,m} s_g^H + N_m.$$  

(33)

Since $s_g$ is orthogonal to $s_{g'}$, $g \neq g'$, the signals sent by group $g$ can be separated as

$$y_{g,m} = \frac{1}{G \sigma_p^2} Y_{m,s_g} = \sum_{k \in G_g} h_{k,m} + \frac{1}{G \sigma_p^2} N_m s_g$$

$$= D_{\mathcal{Q}} r_{m,s} + \tilde{n}_m,$$  

(34)

where $D_{\mathcal{Q}} = [\Phi(\rho_1)_{1,\mathcal{Q}1}, \Phi(\rho_2)_{1,\mathcal{Q}2}, \ldots]$, $r_{m,s} = [r_{1,m}]_{\mathcal{Q}1}, [r_{2,m}]_{\mathcal{Q}2}, \ldots]^H$, and $\tilde{n}_m = \frac{1}{\sqrt{G}} N_m s_g$ is the equivalent Gaussian noise vector.
Define
\[
\alpha^\ast = \text{blkdiag}\{\text{diag}(\alpha_1, \alpha_1, \ldots), \text{diag}(\alpha_2, \alpha_2, \ldots), \ldots\},
\]
\[
\Lambda^\ast = \text{blkdiag}\{[\Lambda_1, \Lambda_1, \ldots], [\Lambda_2, \Lambda_2, \ldots], \ldots\},
\]
and \(v_{mQ}^\ast = [v_{mQ_1}, v_{mQ_2}, \ldots]^H\), where the components of \(v_{mQ}^\ast\) are the dimension-reduced process noise vector. Then we can derive the following space-domain model for the tracking of the dimension-reduced virtual channel \(r_{mQ}\):
\[
r_{mQ} = \alpha^\ast r_{mQ} + v_{mQ}^\ast.
\]
\[
y_{m\lambda} = D_{\lambda} r_{mQ} + \tilde{n}_{k,m}.
\]
Obviously, the above space-domain is linear, and the Kalman filter can be utilized to track the virtual channels.

IV. DOWNLINK CHANNEL MODEL RECONSTRUCTION AND CHANNEL RESTORATION

Similar to the UL case, the DL channel between the BS and the \(k\)-th user during the \(m\)-th time block can be expressed as:
\[
g_{k,m} = \int_{-\infty}^{+\infty} \sum_{l=1}^{L} a(\varphi_{k,l,m}) e^{j2\pi f_{\lambda} l} T_s r_{k}(\varphi_{k,l,m}, \nu) d\nu,
\]
where \(\varphi\) is the direction of departure (DOD) the propagation path. Similar to (6), the DL channel model \(g_{k,m}\) can be arranged as the VCR with DL spatial signatures \(Q_k\), i.e.,
\[
g_{k,m} = \Phi(\rho_k^H) \text{diag}(\rho_k^H) \text{diag}(\rho_k^H) r_{k,m}^H = \Phi(\rho_k^H) \text{diag}(\rho_k^H) Q_k^H r_{k,m}^H.
\]

A. DL Channel Model Parameters Reconstruction

In the FDD mode, since the channel covariance matrices between UL and DL have no reciprocity, the DL model parameters \(X_k = \{\rho_k^H, c_k, \alpha_k, \Lambda_k', \sigma_{n,k}\}^H\) are not the same to the UL ones. Thanks to the angle reciprocity, we can reconstruct partial parameters in \(X_k\). However, \(\Lambda_k', \sigma_{n,k}\) are closely related with the carrier frequency, and can not be perfectly inferred from the UL. An alternative method is to relearn those parameters, which will need some dedicated training and waste the system bandwidth. Thus, we will resort to the optimal Bayesian Kalman filtering (OBKF) to implement both the effective channel tracking and the restoration of the model parameters. We will see that this method does not need dedicated training period, and will ensure the real-time channel updating. In the following, we will first introduce the reconstruction of \(\alpha_k', \rho_k^H\). Then, in the next subsection, the OBKF will be given.

1) \(\alpha_k': \) It is obvious that the moving velocities don’t change between UL and DL. Thus, the Doppler frequency \(\nu_k^\max\) in the DL can be obtained from \(\lambda, \lambda'\) and \(\nu_k^\max\) as \(\nu_k^\max = \frac{\lambda}{\lambda'} \nu_k^\max\). Then, \(\alpha_k'\) is given by
\[
\alpha_k' = J_0(2\pi \nu_k^\max L_s T_s).
\]
where \(J_0(\cdot)\) is the zeroth-order Bessel function of the first kind.

2) \(Q_k'\) and \(\rho_k^H\): As there’s reciprocity lying in the propagation paths, the DODs of DL paths is the same as the DOAs of UL paths at the BS. With this observation, we can recover \(Q_k'\) as well as \(\rho_k^H\) from \(Q_k\) and \(\rho_k\). Similar to (4), we have
\[
\sin(\theta_k) = \frac{\nu_k^\max}{N_d} = \frac{\lambda'}{\lambda} \frac{\nu_k^\max}{N_d}.
\]
Then, it can be obtained that
\[
(p' + [\rho_k^H]_{p'}) = \frac{p + [\rho_k^H]_{p'}}{\lambda'}, \quad p \in Q_k,
\]
where \(p' = (p + [\rho_k^H]_{p'})\) includes all the \(p'\) that satisfies (41). Notice that different \(p \in Q\) may be mapped on the same grid \(p'\) in the DL virtual channel. If two rays in the UL are mapped on the same grid \(p'\) with different bias \(\rho'\), our scheme is to see them as one ray and adopt the average of their bias. For example, if the bias of two specific ray is 0.1 and 0.3, respectively, we regard them as the very ray with the bias 0.2. Furthermore, the corresponding \(c_k\) can be determined by \(Q_k'\), as \(c_k'_{i,j} = 1\) when \(i \in Q_k'\).
B. DL Channel Restoration by Optimal Bayesian Kalman Filtering

Now, we aims to track \([\hat{g}_{k,m}]_{\mathcal{Q}'}\) with the reconstructed partial knowledge about \(\mathbb{E}_P\) in the previous subsec- tion. Similar to (32), the users are divided into \(G'\) groups based on their DL spatial signatures, i.e.,

\[
c'_k c'_j^T = 0, \quad k \neq j. \tag{42}
\]

The user indices of the group \(g\) are collected into the set \(G'\). Due to the difference of spatial signatures, users in one group can utilize the same training sequences. Therefore, to estimate \([Q']_k\) coefficients, \([Q']_k\) orthogonal training sequences are required. So we derive a \(M_g \times M_g\) matrix \(T_g\) satisfying \(T_g^H T_g = M_g \sigma_g^2 I_{M_g}\), \(M_g = \max_{k \in G'}(Q'_k)\) and denote \(S_k = [T_g^H]_{k\in G'}: \) as the training sequences for the \(k\)-th user. Then, \(S_k\) is allocated to the beam \([\Phi(p_k)^H]_{k\in G'}\). Since the training sequences are transmitted for users in the same group simultaneously, the transmitted signals for DL channel estimation of group \(g\) can be given by \(\Gamma_g = \sum_{k \in G'} [\Phi(p_k)^H].\) The received signal of the \(k\)-th group in user is given by

\[
y_{k,m}^H = g_{k,m}^H \Gamma_g + n_{k,m}^H = [g_{k,m}]_{Q'_k} S_k + n_{k,m}^H. \tag{43}
\]

To eliminate the inter-group interference, we can further derive that

\[
\tilde{y}_{k,m}' = \frac{1}{M_g \sigma_p^2} S_k y_{k,m} = [g_{k,m}]_{Q'_k} \hat{n}_{k,m} + n_{k,m}', \tag{44}
\]

where the equivalent complex Gaussian noise vector \(\hat{n}_{k,m}' = \frac{1}{\sigma_p^2} S_k n_{k,m} \sim CN(0, \frac{\sigma_p^2}{\sigma_p^2} I_{Q'_k})\). As mentioned in the previous subsection, the statistics of the noise \(\sigma_p^2\) and \([A'_k]_{Q'_k}\) are unknown. For one specific user, we denote \(\theta = \{\sigma_n^2, \{A'_k\}_{j \in Q'}\}\) as the set of all the unknown parameters, and use the superscript \(\theta\) to express that the unknown parameters relate partly or wholly with it. Then we can obtain the DL state-space model as

\[
\begin{align*}
[\hat{g}_{k,m}]_{Q'_k} &= \alpha' [g_{k,m}]_{Q'_k} + [n_{k,m}']_{Q'_k}, \\
[y_{k,m}']_{Q'_k} &= [g_{k,m}]_{Q'_k} + [n_{k,m}']_{Q'_k}.
\end{align*} \tag{45}
\]

Thus, it is unable to track the DL channel by using the classical KF, whose performance is very sensitive to the accuracy of noise statistics. Nonetheless, there are many robust KF methods to handle this problem, such as IBF KF in [23]. In order to fully utilize the additional information in the observed signal, the OBKF method will be adopted for our DL channel tracking process. The process is divided into 3 parts: the OBKF process, the sum-product algorithm for posterior noise statistics, and the MCMC computation.

1) OBKF for DL Channel Tracking: Since each user can track their channels and restore the model parameters independently, we will ignore the subscript \(k\) in the following for simplicity.

Hence, under the OBKF framework, the following equations can be utilized to effectively track the DL virtual channel \([\hat{g}_{k,m}]_{Q'_k}\) as

\[
\begin{align*}
\hat{z}_{m}' &= \hat{y}_{m}' - [\hat{g}_{k,m}]_{Q'_k}, \\
K_m &= \mathbb{E}[F_m | \hat{y}_{m}' - [\hat{g}_{k,m}]_{Q'_k}], \\
&= \mathbb{E}^{-1} \{[P_m + \frac{\sigma_p^2}{\sigma_n^2} I_{Q'_k}] | \hat{y}_{m}' - [\hat{g}_{k,m}]_{Q'_k}\}, \tag{46}
\end{align*}
\]

\[
\begin{align*}
\hat{[g}_{m+1}]_{Q'_k} &= \alpha'[\hat{g}_{m}]_{Q'_k} + \alpha' K_m \hat{z}_{m}', \\
\mathbb{E}[P_{m+1}^{|y_{m}'(m-1)}] &= \alpha'^2 (I - K_m \Theta) \mathbb{E}[P_m^{|y_{m}'(m)}] + \mathbb{E}[\{A'_k\}^{|y_{m}'(m)}]. \tag{47}
\end{align*}
\]

where \(\hat{y}_{m}' = [\hat{y}_{1}' H, \hat{y}_{2}' H, \ldots, \hat{y}_{m}' H]\), and \(P_m^{|y_{m}'(m)} = \mathbb{E}\{(g_{m}^H - [\hat{g}_{m}]_{Q'_k}) (g_{m}^H - [\hat{g}_{m}]_{Q'_k})^H\}\) is the covariance matrix of the channel estimation error relative to \(\theta\) at block \(m\).

In order to reduce the computation complexity, in simple case, we make the approximation \(\mathbb{E}[P_{m+1}^{|y_{m}'(m-1)}] \approx \mathbb{E}[P_m^{|y_{m}'(m)}]\), and replace \(\mathbb{E}[P_m^{|y_{m}'(m-1)}]\) from the previous iteration [24]. This option is more efficient computationally, which is due to the fact that we do not require to do all the recursions in (46)–(49) at each block \(m\).

From (46)–(49), we will find that two noise condition expectations should be evaluated with respect to the posterior distribution \(p(\theta | y_{m}'(m)) \propto p(y_{m}'(m) | \theta)p(\theta)\), where \(p(y_{m}'(m) | \theta)\) is a likelihood function of \(\theta\) given the observation \(y_{m}'(m)\). Since there may be no closed-form solution for \(p(\theta | y_{m}'(m))\), to implement the OBKF process, we resort to the MCMC method to derive samples with the knowledge of the posterior distribution \(p(\theta | y_{m}'(m))\) and to approximate \(\mathbb{E}[\{A'_k\}^{|y_{m}'(m)}]\) as the mean of these samples. With the Bayes rule, it can be confirmed that the likelihood function \(p(y_{m}'(m) | \theta)\) should be calculated to determine \(p(\theta | y_{m}'(m))\).

With (45) and the property of the the Markov model, we can obtain

\[
\begin{align*}
\mathbb{E}[y_{m}'(m-1) | x'(m); \theta] &= p(y_{m}'(m) | x'(m); \theta) = \mathcal{CN}_N \left(y_{m}'(m), \sigma_p^2 I_{Q'_k} \right), \\
p([g_{m+1}]_{Q'_k}, x'(m); \theta) &= \mathcal{CN} \left([g_{m+1}]_{Q'_k}, \sigma_n^2 I_{Q'_k} \right), \tag{50}
\end{align*}
\]

where \(x'(m) = [g_{1}^H, g_{2}^H, \ldots, g_{m}^H]_{Q'_k}\) is set of the past \(m\) \([g_{m}]_{Q'_k}\).

With (50) and (51), the marginalization of \(p(y_{m}'(m) | \theta)\) can be factorized as (52), on the top of the next page.

Then, \(p(y_{m}'(m) | \theta)\) can be denoted with a factor graph, as shown in Fig. 4, where the factors in (52) are represented by “functions nodes” marked blue and red boxes, and the corresponding random variables are represented by “variable nodes” marked as green circles. One specific variable node \(x\) connects with the function nodes \(f\), whose augments contain \(x\). Furthermore, we will resort to the belief propagation (BP),
also known as sum-product message passing, to implement the message-passing in our constructed factor graph. Fig. 4. BP passes real valued messages along the edges between nodes in the factor graph. Specifically, for the function node \( f \) and the variable node \( x \), the messages from \( f \) to \( x \) and from \( x \) to \( f \) are separately defined as \( \Omega_{f \rightarrow x}(x) \) and \( \Omega_{x \rightarrow f}(x) \), whose augments is \( x \). With the BP theory, we can obtain

\[
\Omega_{x \rightarrow f}(x) = \prod_{f' \in \mathcal{N}(x)/f} \Omega_{f' \rightarrow x}(x),
\]

(53)

\[
\Omega_{f \rightarrow x}(x) = \int_{x'} \left( f(x) \prod_{x' \in \mathcal{N}(f)/x} \Omega_{x' \rightarrow f}(x') \right),
\]

(54)

where the set \( \mathcal{N}(x) \) collects all the neighbouring nodes of the given node \( x \) in one factor graph, and \( \sim \) \( x \) possesses the same meaning with the same notation \[35\].

2) Sum-Product Algorithm for Posterior Noise Statistics: A node in the factor graph operates when the node receives all messages from its neighbouring nodes. The first step to run a factor graph is that all leaf function node sends its message to its neighbouring nodes. For expression simplicity, we define the factor nodes and variable nodes in Fig. 4 as \( w_i = [\tilde{g}_i]_Q \), \( f_{A, i} = \mathcal{C}N(w_i; \alpha'w_{i-1}, [\Lambda']_Q) \), and \( f_{B, i} = \mathcal{C}N(y_i; \mu, \Sigma) \), \( i = 1, \ldots, m \), while \( f_{A, 1} = p(w_1) = \mathcal{C}N(w_1; 0, \Lambda_1^Q) \).

It can be seen from Fig. 4 that there’s three kind of message in the factor graph, i.e., \( \Omega_{f_{A, i} \rightarrow w_i} \), \( \Omega_{f_{B, i} \rightarrow w_i} \), and \( \Omega_{w_i \rightarrow f_{A, i}} \). Since, we only need to consider the forward passing message, the expression of \( \Omega_{w_i \rightarrow f_{A, i}} \) can be omitted here. With (50), (51), and (53)-(54), it can be readily checked from Fig. 4 that

\[
\Omega_{f_{A, i} \rightarrow w_i} = \mathcal{C}N(\tilde{y}_i; w_i, \frac{\sigma_n^2}{\sigma_p^2}\Gamma_{i}^{Q_i'}). 
\]

(55)

With respect to the term \( \Omega_{f_{A, i} \rightarrow w_i} \), we have the following lemma.

**Lemma 1:** For all \( 1 \leq i \leq m - 1 \), the message \( \Omega_{f_{A, i+1} \rightarrow w_{i+1}} \) in Fig. 4 can be expressed as:

\[
\Omega_{f_{A, i+1} \rightarrow w_{i+1}} = \omega_{i+1} \mathcal{C}N(w_{i+1}; \mu_{i+1}, \Sigma_{i+1}),
\]

(56)

where

\[
\Sigma_{i+1} = \left( \left[ \Lambda' \right]_{Q_i'}^{-1} - \alpha'^2 \left[ \Lambda' \right]_{Q_i'}^{-1} \Gamma_i \left[ \Lambda' \right]_{Q_i'}^{-1} \right)^{-1},
\]

(57)

\[
\mu_{i+1} = {\alpha'} \Sigma_{i+1} \left[ \Lambda' \right]_{Q_i'}^{-1} \Gamma_i \left( \left( \frac{\sigma_n^2}{\sigma_p^2} \left[ \Lambda' \right]_{Q_i}^{-1} \right)^{-1} \tilde{y}_i + \Sigma_{i}^{-1} \mu_{i} \right),
\]

(58)

where \( \Gamma_i \) and \( \mu_i \) are defined in the following proof part. Furthermore, in every step \( i \), the parameters \( \omega_{i+1}, \mu_{i+1}, \) and \( \Sigma_{i+1} \) in \( \Omega_{f_{A, i+1} \rightarrow w_{i+1}} \) are only related to those in \( \Omega_{f_{A, i+1} \rightarrow w_i} \). In addition, it is checked that \( \Omega_{f_{A, i+1} \rightarrow w_i} = \omega_{i+1} \mathcal{C}N(w_{i+1}; \mu_{i+1}, \Sigma_{i+1}) \).

**Proof:** Refer to Appendix A.

With Lemma 1, we will finally obtain the message \( \Omega_{f_{A, 1} \rightarrow w_m} = \omega_m \mathcal{C}N(w_m; \mu_m, \Sigma_m) \). Then the equation (52) can be rewritten as:

\[
p(\tilde{y}'(m)|\vartheta) = \frac{1}{\Omega_{f_{A, 1} \rightarrow w_m} \Omega_{f_{B, 1} \rightarrow w_m} dw_m} \int_{w_m} \mathcal{C}N(\tilde{y}_i'; w_i, \frac{\sigma_n^2}{\sigma_p^2}\Gamma_{i}^{Q_i'}) \mathcal{C}N(w_m; \mu_m, \Sigma_m) dw_m
\]

(59)

where

\[
\Delta_m^{-1} = \left( \frac{\sigma_n^2}{\sigma_p^2} I_{[\vartheta]} \right)^{-1} + \Sigma_m^{-1},
\]

(60)

\[
\mathcal{G}_m = \Delta_m \left( \frac{\sigma_n^2}{\sigma_p^2} I_{[\vartheta]} \right)^{-1} \tilde{y}_m + \Sigma_m^{-1} \mu_m.
\]

(61)

Hence, according to (59), the likelihood function \( p(\tilde{y}'(m)|\vartheta) \) can be derived, where all the parameters defined before can be obtained according to the above recursion processes.

3) MCMC Computation: As the two posterior effective noise statistics \( \mathbb{E}_\vartheta \left\{ \frac{\sigma_n^2}{\sigma_p^2} I_{[\vartheta]} \mid \tilde{y}'(m) \right\} \) and \( \mathbb{E}_\vartheta \left\{ [\Lambda']_{Q_i'} \mid \tilde{y}'(m) \right\} \) are unknown, we employ the Metropolis Hastings MCMC algorithm to estimate them. This algorithm is used to the case where the proposal distribution is no longer a symmetric function of its arguments \[36\]. At the \( j \)-th iteration, the accepted MCMC sample \( \vartheta^{(j)} \) is generated. A MCMC sample \( \vartheta \) to be choose will be later generated according to a proposal distribution \( p(\vartheta|\vartheta^{(j)}) \). As the specific choice of proposal distribution can have a prominent effect on the performance of the algorithm, we choose a Gaussian distribution centred on
the current state $\theta^{(j)}$. The candidate sample $\hat{\theta}$ will be chosen to be either accepted or rejected with respect to an ratio $r$ defined as

$$r = \min \left\{ 1, \frac{p(\theta^{(j)}|\hat{\theta})p(\hat{\theta}|m)}{p(\hat{\theta}|\theta^{(j)})p(\theta^{(j)}|m)} \right\}. \quad (62)$$

The $(j+1)$-th MCMC sample is

$$\theta^{(j+1)} = \begin{cases} \hat{\theta} & \text{with probability } r, \\ \theta^{(j)} & \text{otherwise.} \end{cases} \quad (63)$$

We can iterate the process in (62), (63), and achieve a sequence of MCMC samples. After generating a scale of samples, the two posterior noise statistics can be derived as the mean of the sequence of accepted samples.

**Remark 2:** In the UL process, the computational complexity is closely related with the derivation of $R_{l,m}^{(i-1)}$, $\Lambda_{l,m}^{(i-1)}$, and $\mathbf{P}_{k,m}^{(i-1)}$ in the E-step, where the KF and RTSS are utilized in our previous work [21]. As pointed out in [21], the complexities of KF and RTSS are $O(M_dN_t^3)$ and $O(M_dN_t)$, respectively. The computation complexity of Algorithm 2 is dominantly determined by the proposed OBKF method, where MCMC and the sum-product algorithm are introduced in the inner recursion. Before proceeding, let us define $M_d$, $N_{MC}$ and $N_t$ as the number of blocks within OBKF, the number of MCMC iterations and the number of non-zero points in the DL virtual channel, respectively. Since all the matrices in OBKF are diagonal, the computational complexity of this process is $O(M_dN_t^2)$. Moreover, the computational complexity of the sum-product algorithm and the MCMC method are $O(M_d^2N_{MC}N_t^4)$ and $O(M_dN_{MC})$, respectively. Then, through omitting some small order terms, the overall computational complexity including UL and DL is $O(M_dN_t^3 + M_d^2N_{MC}N_t^4Q)$. The steps of the whole procedure for the DL channel reconstruction and restoration are summarized in Algorithm 2.

**V. SIMULATIONS RESULTS**

In this section, we will evaluate the performance of our proposed tracking scheme through numerical simulation. We consider a massive MIMO network where the BS is equipped with $N_t = 128$ antennas. $K = 32$ is the number of users, while they are divided into 8 groups. The length of training sequences $L_t = 4$. We take one group as an example to show the perfect performance. The simulation parameters are listed in TABLE I. The signal-to-noise ratio $SNR = \sigma_p^2/\sigma_n^2$.

The performance is measured as the average MSEs of the model parameters as well as the virtual channel, i.e.,

$$MSE_x = \frac{1}{\tau} \sum_{i=1}^{\tau} ||x_i - \hat{x}_i||^2, \quad x = \alpha, \Lambda, \rho, \sigma_n^2, \hat{h}, \hat{g}. \quad (64)$$

We first investigate the convergence of the UL EM process. Fig. 5 shows the MSEs versus the number of EM iterations. $M_d = 15$ channel blocks are utilized to learn model parameters. We can see from Fig. 5 that after 5 iterations, all the parameters have arrived at their steady states, which shows that the algorithm converges very fast.

Fig. 6(a) presents the MSE of different model parameters capturing versus SNR, with EM algorithm running 5 iterations for each SNR case. With the increase of the SNR, it shows that the MSE of all parameters decrease almost linearly. Moreover, we show the performance of the estimation for the off-grid bias and spatial signature performance in Fig. 6(b), with $SNR = 20dB$. They are also estimated very accurately.

After UL capture of all parameters and DL reconstruction of partial parameters, the next step is to track the DL channel by adopt OBKF, with the known parameters, meanwhile restore
Algorithm 2 UL Learning Aided DL Reconstruction and Restoration

1: input: \( \mathbf{y}, p(\theta), \mathbf{y}'(m) \).
2: initialize: \( \hat{\alpha}(0), \hat{\lambda}(0), \hat{\rho}(0), \sigma^2_0, \theta(0) \sim \pi(\theta) \).
3: for \( l = 0, 1, \ldots, l \) do
4: \( \hat{c}^{(l)} \leftarrow \text{Algorithm 1}, \{p_k^{(l)}\} \leftarrow (28) \) and the constraint \( \{p_k\} \subseteq [-\frac{1}{2}, \frac{1}{2}] \).
5: \( \hat{\alpha}^{(l)} \leftarrow (29), \{\hat{\alpha}_k\} \leftarrow (30), \sigma^2_n^{(l)} \leftarrow (31) \).
6: end for
7: \( \alpha_q^{(l)} \leftarrow (39), Q_k^{(l)} \leftarrow (41) \).
8: \( [\hat{g}_0]_{\theta} \leftarrow \mathbb{E} \{[\hat{g}_m]_{\theta}^q\}, \mathbb{E} \{[\hat{p}]_{\theta}^q|\hat{y}(0)|\} \leftarrow \mathbb{E} \{[\hat{m}]_{\theta}^q\} \).
9: \( \mathbb{E} \{[\hat{y}'(0)]_{\theta} \} \leftarrow \mathbb{E} \{[\hat{y}'(0)]_{\theta}^q\} \).
10: \( \mathbb{E} \{[\hat{y}'(m-1)]_{\theta} \} \leftarrow \mathbb{E} \{[\hat{y}'(m-1)]_{\theta}^q\} \).
11: \( \hat{\vartheta}^{(l)} \leftarrow p(\hat{\vartheta}^{(l-1)}) \).
12: for \( m = 2, 3, \ldots, m \) do
13: \( \nu_j \leftarrow \left( \frac{\sigma^2_q}{\sigma^2_q}\right)_{\theta}^{-1} \hat{y}_j + \Sigma_j^{-1} \mu_j \).
14: \( \Gamma_j \leftarrow \left( \alpha^2_{\theta} \right)_{\theta}^{-1} \hat{y}_j + \Sigma_j^{-1} \mu_j \).
15: \( \Sigma_{\theta} \leftarrow \left( \alpha^2_{\theta} \right)_{\theta}^{-1} \hat{y}_j + \Sigma_j^{-1} \mu_j \).
16: \( \mu_{\theta} \leftarrow \left( \alpha^2_{\theta} \right)_{\theta}^{-1} \Gamma_j \left( \frac{\sigma^2_q}{\sigma^2_q} \right)_{\theta}^{-1} \hat{y}_j \).
17: end for
18: if \( \nu_j \) then
19: \( \omega_j \leftarrow \text{using (58)} \).
20: \( \Delta_m \leftarrow \left( \frac{\sigma^2_q}{\sigma^2_q}\right)_{\theta}^{-1} + \Sigma_m^{-1} \).
21: \( G_m \leftarrow \Delta_m \left( \frac{\sigma^2_q}{\sigma^2_q} \right)_{\theta}^{-1} \hat{y}_j + \Sigma_m^{-1} \mu_m \).
22: \( p(\hat{y}'(m)) \leftarrow \text{using (59)} \).
23: \( p(\hat{y}'(m)) \leftarrow \text{using (59)} \).
24: \( r \leftarrow \min \left\{ \frac{p(\hat{y}'(m))|\hat{y}'(m)|}{p(\hat{y}'(m))|\hat{y}'(m)|}, \zeta \sim \text{unif}(0, 1) \right\} \).
25: if \( \zeta < r \) then
26: \( \hat{\vartheta}^{(l)} \leftarrow \hat{\vartheta}^{(l-1)} \).
27: end if
28: end for
29: \( \hat{\vartheta}^{(l)} \leftarrow \text{using (58)} \).
30: \( \mathbb{E} \left\{ C_\theta \right\}_{\theta}^q \left\{ \hat{y}'(m) \right\}_{\theta} \).
31: \( \mathbb{E} \left\{ C_\theta \right\}_{\theta}^q \left\{ \hat{y}'(m) \right\}_{\theta} \right\} \leftarrow \{\hat{\vartheta}^{(1)}, \hat{\vartheta}^{(2)}, \ldots, \hat{\vartheta}^{(m)}\} \).
32: \( \mathbb{E} \left\{ C_\theta \right\}_{\theta}^q \left\{ \hat{y}'(m) \right\}_{\theta} \).
33: \( \mathbb{E} \left\{ C_\theta \right\}_{\theta}^q \left\{ \hat{y}'(m) \right\}_{\theta} \).
34: return \( [\hat{g}_m]_{\theta} \).

the unreconstructed parameters for later tracking. To decrease the computation complexity, we will adopt OBKF for a limited number of time-blocks, and then use classical KF to continue tracking the channel.

First, we studies the MSE of the two unknown DL channel model parameters at the last OBKF time-block versus the number of time-block using OBKF, with different SNR, and the MSE versus SNR with different number of OBKF time-block. In Fig. 7, we can see that the MSE of the two unknown DL model parameters decreases in each SNR case, and almost arrive at their convergence point when OBKF time-block \( M_d = 15 \). As SNR goes higher, the convergence point can be arrived when OBKF time-block \( M_d = 10 \). It also shows that the curves decrease nearly linearly with the increase of the SNR, in Fig. 8.

We can find that the performance are better when SNR is higher. We can explain the above phenomenon that OBKF
is not only restoring the virtual channel, but also restoring the unknown parameters. And after a scale of restoring time, the parameters will be very close to the true one, so the estimated virtual channel will have a good performance.

Then we studies the MSE of virtual channel for each time-block, including both the OBKF time-blocks and the later classical KF time-blocks, with different SNRs, as shown in Fig. 9. The traditional LS estimation with perfect spatial characteristics (denoted as “LS”), the classical KF with perfect parameters (denoted as “perfect KF”), and classical KF with weak parameters (denoted as “weak KF”) are compared with our proposed method (denoted as “OBKF”). We set the number of OBKF time-block $M_d = 10$, with which we can obtain almost the best performance. From Fig. 9 we can obtain that the MSE of tracked virtual channel decreases when OBKF runs. We can see that the performance converges fast to be steady and is very close to the performance of “perfect KF” at $m_d = 6$. Furthermore, our proposed method outperforms “LS” significantly. The above observations show the accuracy of our method.

To illustrate the performance of the method more completely, Fig. 10 shows the performance of the four kinds of methods versus the SNR. From Fig. 10, we can see that the MSE of “LS” and “weak KF” is far away from that of “perfect KF”, while our “OBKF” has a much better performance than “LS” as well as “weak KF”. Moreover, with the SNR increasing, the difference among the MSE curves decreases very fast. Especially, at SNR = 30, the curves are almost the same. Notice that the gap between our method and weak KF is also decreasing. This can be explained as follows. At low SNR, our method obtains a huge gain by utilizing the correlation between neighboring channel blocks. But with SNR increasing, the performance is mostly decided on SNR, meanwhile the effect of correlation is diminishing. In addition, the gap between our method and LS is still large at high SNR.

Finally, we show the MSE of the two noise parameters versus SNR for different velocity at the last OBKF time-block, while $M_d = 15$. In Fig. 11, we can find that the performance is better at slower velocity, while at higher velocity the performance is only a little worse and is acceptable.

**VI. CONCLUSION**

In this paper, we proposed a skillful scheme for the DL channel tracking. First, with the help of VCR, a dynamic uplink (UL) massive MIMO channel model was built with the consideration of off-grid refinement. Then, a coordinate-wise maximization based expectation maximization (EM) algorithm was adopted in the model parameters learning period. Thanks to the angle reciprocity, with the knowledge of UL channel model parameters, we recovered some of the parameters of DL channel model. After that, as there remains some parameters which could not be perfectly inferred from the UL ones, we resorted to OBKF method to accurately track the DL channel. During the method, factor-graph and Metropolis Hastings MCMC were applied to track the expectation of posterior statistics. Numerical results showed that our proposed scheme
has not only a strong convergence, but also a very low estimation MSE.

**APPENDIX A**

**PROOF OF LEMMA 1**

Before proceeding, we give the following property:

\[ CN(w_{i+1}; \alpha' \omega_i, [\Lambda']_{Q'}) = \frac{1}{|\pi[\Lambda']_{Q'}|} \exp \left( - (w_{i+1} - \alpha' \omega_i)^H [\Lambda']^{-1}_{Q'} (w_{i+1} - \alpha' \omega_i) \right) \]

\[ = \alpha'^{-2}Q'CN(w_i; w_{i+1}/\alpha', [\alpha'^{-2}\Lambda']_{Q'}). \]  \hfill (65)

With the above equation, if \( \Omega_{f_{A,i},-w_i} = \omega_iCN(w_i; \mu_i, \Sigma_i) \) holds for \( i \geq 1 \), we can derive

\[ \Omega_{f_{A,i+1},-w_{i+1}} = \int f_{A,i+1} \Omega_{f_{A,i},-w_i} \Omega_{f_{B,i},-w_i} dw_i \]

\[ = \omega_i \int CN(w_{i+1}; \alpha' \omega_i, [\Lambda']_{Q'}) \times CN\left( y_i'; w_i; \frac{\sigma^2_i}{\sigma^2_p} I_{[Q']} \right) CN(w_i; \mu_i, \Sigma_i) dw_i \]

\[ = \omega_i \alpha'^{-2}Q'CN\left( 0; y_i'; \frac{\sigma^2_i}{\sigma^2_p} I_{[Q']} \right) CN(0; \mu_i, \Sigma_i) \]

\[ \times \frac{CN(0; \alpha'^{-1}w_{i+1}, [\alpha'^{-2}\Lambda']_{Q'})}{CN(0; \Gamma_i; \mu_{i+1}, \Sigma_{i+1})}. \]  \hfill (66)

where

\[ \nu_i = \left( \frac{\sigma^2_i}{\sigma^2_p} I_{[Q']} \right)^{-1} y_i + \Sigma_i^{-1} \mu_i, \]  \hfill (67)

\[ \Gamma_i = \left( \alpha'^2[\Lambda']_{Q'}^{-1} + \left( \frac{\sigma^2_i}{\sigma^2_p} I_{[Q']} \right)^{-1} + \Sigma_i^{-1} \right)^{-1}, \]  \hfill (68)

and (79) in the Appendix B are utilized in the above derivations.

Furthermore, with respect to

\[ CN(0; \alpha'^{-1}w_{i+1}; [\alpha'^{-2}\Lambda']_{Q'}) \]

in (66), we can obtain, (69) and (70), as shown at the top of the next page, where

\[ \Sigma_{i+1} = \left( [\Lambda']_{Q'}^{-1} - \alpha'^2[\Lambda']_{Q'}^{-1} \Gamma_i [\Lambda']_{Q'}^{-1} \right)^{-1}, \]  \hfill (71)

\[ \mu_{i+1} = \alpha \Sigma_{i+1} [\Lambda']_{Q'}^{-1} \Gamma_i \left( \frac{\sigma^2_i}{\sigma^2_p} I_{[Q']} \right)^{-1} y_i + \Sigma_i^{-1} \mu_i. \]  \hfill (72)

Notice that the equations (79) and (81) in the Appendix B, and the following properties are utilized in the above derivations.

\[ CN(0; -\alpha^{-1}[\Lambda']_{Q'} \nu_i, -\alpha'^{-2}[\Lambda']_{Q'} \Gamma_i^{-1}[\Lambda']_{Q'}) \]

\[ = \frac{|\Gamma_i^{-1}|}{|\alpha'^{-2}[\Lambda']_{Q'} \Gamma_i^{-1}[\Lambda']_{Q'}|} CN\left( \nu_i; 0, \Gamma_i^{-1} \right), \]  \hfill (73)

\[ CN(0; \alpha \Sigma_{i+1} [\Lambda']_{Q'}^{-1} \Gamma_i \nu_i, \mu_i, \Sigma_i) = \frac{1}{|\Sigma_{i+1}|} \exp \left( - \alpha'^2 \nu_i^H \Gamma_i [\Lambda']_{Q'}^{-1} \Sigma_{i+1} [\Lambda']_{Q'}^{-1} \Gamma_i \nu_i \right) \]

\[ = \frac{|\alpha'^{-2} \Gamma_i^{-1}[\Lambda']_{Q'} \Sigma_{i+1} [\Lambda']_{Q'}^{-1} \Gamma_i^{-1}|}{|\Sigma_{i+1}|} \times CN(\nu_i; 0, \alpha'^{-2} \Gamma_i^{-1}[\Lambda']_{Q'} \Sigma_{i+1} [\Lambda']_{Q'}^{-1} \Gamma_i^{-1}). \]  \hfill (74)

So, \( \Omega_{f_{A,i+1},-w_{i+1}} \) can be reexpressed from (66) and (69) as:

\[ \Omega_{f_{A,i+1},-w_{i+1}} = \omega_i \alpha'^{-2}Q'CN\left( 0; y_i', \frac{\sigma^2_i}{\sigma^2_p} I_{[Q']} \right) CN(0; \mu_i, \Sigma_i) \]

\[ \times \frac{CN(0; \alpha'^{-1}w_{i+1}, [\alpha'^{-2}\Lambda']_{Q'})}{CN(0; \Gamma_i; \mu_{i+1}, \Sigma_{i+1})}. \]

\[ = \omega_i \alpha'^{-2}Q'CN\left( 0; y_i', \frac{\sigma^2_i}{\sigma^2_p} I_{[Q']} \right) CN(0; \mu_i, \Sigma_i) \]

\[ \times \frac{CN(0; \alpha'^{-1}w_{i+1}, [\alpha'^{-2}\Lambda']_{Q'})}{CN(0; \Gamma_i; \mu_{i+1}, \Sigma_{i+1})}. \]

\[ \times \left| [\Lambda']_{Q'} \right| CN\left( \nu_i; 0, (\Gamma_i + \alpha'^2 \Gamma_i [\Lambda']_{Q'}^{1} \Sigma_{i+1} [\Lambda']_{Q'}^{1} \Gamma_i^{-1})^{-1} \right). \]  \hfill (75)

where

\[ \omega_i \]

\[ = \omega_iCN\left( 0; y_i', \frac{\sigma^2_i}{\sigma^2_p} I_{[Q']} \right) CN(0; \mu_i, \Sigma_i) \]

\[ \times \frac{|\Gamma_i| |\Sigma_{i+1}| \times (\Gamma_i + \alpha'^2 \Gamma_i [\Lambda']_{Q'}^{1} \Sigma_{i+1} [\Lambda']_{Q'}^{1} \Gamma_i^{-1})^{-1}}{|[\Lambda']_{Q'}| CN(\nu_i; 0, (\Gamma_i + \alpha'^2 \Gamma_i [\Lambda']_{Q'}^{1} \Sigma_{i+1} [\Lambda']_{Q'}^{1} \Gamma_i^{-1})^{-1})}. \]  \hfill (76)

**APPENDIX B**

**THE PRODUCT OF THE N-DIMENSIONAL COMPLEX GAUSSIAN PDF**

For the \( N \)-dimensional complex Gaussian distribution \( p(x) = CN(x; \mu, \Sigma) \), we can obtain its canonical notation as

\[ p(x) = \frac{1}{|\pi| |\Sigma|} \exp\left(- (x - \mu)^H \Sigma^{-1} (x - \mu) \right) \]

\[ = \exp\left(- \ln |\pi| - \ln |\Sigma| \right) \times \exp\left\{ -x^H \Sigma^{-1} x - \mu^H \Sigma^{-1} \mu + 2 \Re \{ x^H \Sigma^{-1} x \} \right\} \]

\[ = \exp\left( -N \ln |\pi| - \ln |\Sigma| - \mu^H \Sigma^{-1} \mu \right) \]

\[ -x^H \Sigma^{-1} x + 2 \Re \{ x^H \Sigma^{-1} x \}. \]  \hfill (77)

Then, for the PDFs \( p_i(x) = CN(x; \mu_i, \Sigma_i) \), \( i = 1, 2, \ldots, L \), we can derive

\[ \prod_{i=1}^{L} p_i(x) = \prod_{i=1}^{L} CN(x; \mu_i, \Sigma_i) \]

\[ = \exp \left( \sum_{i=1}^{L} \zeta_i - x^H \left( \sum_{i=1}^{L} \Sigma_i^{-1} x \right) \right) \]

\[ + 2 \Re \left\{ x^H \left( \sum_{i=1}^{L} \Sigma_i^{-1} \mu_i \right) \right\}. \]  \hfill (78)

where the term \( \zeta_i = -N \ln |\pi| - \ln |\Sigma_i| - \mu_i^H \Sigma_i^{-1} \mu_i \) is defined in the above equation. Before proceeding, let us define \( \bar{\Sigma}_L = \left( \sum_{i=1}^{L} \Sigma_i^{-1} \right)^{-1} \), and \( \bar{\mu}_L = \bar{\Sigma}_L \left( \sum_{i=1}^{L} \Sigma_i^{-1} \mu_i \right) \).
\[
\begin{align*}
\mathcal{C}N \left( \alpha^{-1}w_{i+1}, \left[ \alpha^{-2} \Lambda' \right]_{Q'} \right) & \quad = \frac{\alpha^4 [Q']^2 \left| \Gamma_i \right|^2}{\left| \Lambda' \right|_{Q'}^2} \mathcal{C}N \left( w_{i+1}; -\alpha^{-1} \Lambda' \nu_i, \alpha^{-2} \left[ \Lambda' \right]_{Q'} \left[ T_i^{-1} \right]_{Q'} \right) \\
& \quad = \alpha^4 [Q']^2 \left| \Gamma_i \right|^2 \mathcal{C}N \left( w_{i+1}; -\nu_i, \alpha^{-2} \Lambda' \left[ T_i^{-1} \right]_{Q'} \right) \\
& \quad = \mathcal{C}N (0, 0, \left[ \Lambda' \right]_{Q'}) \mathcal{C}N (w_{i+1}; \mu_{i+1}, \Sigma_{i+1}) \\
& \quad = \mathcal{C}N \left( \nu_i, 0, \left( \Gamma_i + \alpha^2 \left[ \Lambda' \right]_{Q'} \left[ \Sigma_{i+1} \right]_{Q'} \right) \left[ T_i^{-1} \right]_{Q'} \right) \\
& \quad = \mathcal{C}N \left( \nu_i, 0, \left( \Gamma_i + \alpha^2 \left[ \Lambda' \right]_{Q'} \left[ \Sigma_{i+1} \right]_{Q'} \right) \left[ T_i^{-1} \right]_{Q'} \right) \\
& \quad = \frac{\alpha^2 [Q']^2 \left| \Sigma_{i+1} \right|}{\left| \Lambda' \right|_{Q'}} \\
& \quad = \mathcal{C}N \left( \nu_i, 0, \left( \Gamma_i + \alpha^2 \left[ \Lambda' \right]_{Q'} \left[ \Sigma_{i+1} \right]_{Q'} \right) \left[ T_i^{-1} \right]_{Q'} \right) \\
& \quad = \mathcal{C}N (0; \mu_{i+1}, \Sigma_{i+1}) \mathcal{C}N (0; \mu_{2}, \Sigma_{2}) \\
\mathcal{C}N (x; \mu_{2}, \Sigma_{2}) & \quad = \mathcal{C}N (x; \left( \Sigma_{2} - \Sigma_{1}^{-1} \right)^{-1} \left( \Sigma_{2}^{-1} \mu_{2} - \Sigma_{1}^{-1} \mu_{1} \right), \left( \Sigma_{2} - \Sigma_{1}^{-1} \right)^{-1}) \\
& \quad \times \mathcal{C}N (0; \mu_{1}, \Sigma_{1}) \mathcal{C}N (0; \left( \Sigma_{2} - \Sigma_{1}^{-1} \right)^{-1} \left( \Sigma_{2}^{-1} \mu_{2} - \Sigma_{1}^{-1} \mu_{1} \right), \left( \Sigma_{2} - \Sigma_{1}^{-1} \right)^{-1}) \\
& \quad = \exp \left( \sum_{i=1}^{L} \zeta_{i} - \tilde{\zeta}_{2} \right) \\
& \quad = \exp \left( \sum_{i=1}^{L} \zeta_{i} - \tilde{\zeta}_{2} + \tilde{\zeta}_{L} - x^{H} \Sigma_{L}^{-1} x + 2R \left( x^{H} \Sigma_{L}^{-1} \tilde{\mu}_{L} \right) \right) \\
& \quad = \exp \left( \sum_{i=1}^{L} \zeta_{i} - \tilde{\zeta}_{L} \right) \mathcal{C}N (x; \tilde{\mu}_{L}, \Sigma_{L}), \\
\end{align*}
\]

Hence, the above equation can be reexpressed as

\[
\prod_{i=1}^{L} \mu_{i}(x) = \frac{\mathcal{C}N (0; \mu_{1}, \Sigma_{1}) \mathcal{C}N (0; \mu_{2}, \Sigma_{2})}{\mathcal{C}N (x; \mu_{2}, \Sigma_{2})} = \mathcal{C}N (x; \left( \Sigma_{2} - \Sigma_{1}^{-1} \right)^{-1} \left( \Sigma_{2}^{-1} \mu_{2} - \Sigma_{1}^{-1} \mu_{1} \right), \left( \Sigma_{2} - \Sigma_{1}^{-1} \right)^{-1}) \\
\times \mathcal{C}N (0; \mu_{1}, \Sigma_{1}) \mathcal{C}N (0; \left( \Sigma_{2} - \Sigma_{1}^{-1} \right)^{-1} \left( \Sigma_{2}^{-1} \mu_{2} - \Sigma_{1}^{-1} \mu_{1} \right), \left( \Sigma_{2} - \Sigma_{1}^{-1} \right)^{-1})
\]

where \( \tilde{\zeta}_{L} = -N \ln \pi - \ln | \Sigma_{L} | - \mu_{L}^{H} \Sigma_{L}^{-1} \mu_{L} \).

Specially, for \( L = 2 \), we can obtain (80), on the top of this page.

Moreover, if the terms \( \mu_{1}, \Sigma_{1}, \tilde{\mu}_{2}, \) and \( \tilde{\Sigma}_{2} \) are given, we can derive the quotient of two N-dimensional complex Gaussian PDF (81), on the top of this page.

### References

[1] T. L. Marzetta, “Noncooperative cellular wireless with unlimited numbers of base station antennas,” *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

[2] F. Rusek et al., “Scaling up MIMO: Opportunities and challenges with very large arrays,” *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.

[3] V. Jungnickel et al., “The role of small cells, coordinated multipoint, and massive MIMO in 5G,” *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 44–51, May 2014.

[4] X. Liu, Y. Liu, X. Wang, and H. Lin, “Highly efficient 3-D resource allocation techniques in 5G for NOMA-enabled massive MIMO and relaying systems,” *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2785–2797, Dec. 2017.

[5] W. Tan, M. Matthaiou, S. Jin, and X. Li, “Spectral efficiency of DFT-based processing hybrid architectures in massive MIMO,” *IEEE Wireless Commun. Lett.*, vol. 6, no. 5, pp. 586–589, Oct. 2017.

[6] S. Noh, M. D. Zoltowski, and D. J. Love, “Training sequence design for feedback assisted hybrid beamforming in massive MIMO systems,” *IEEE Trans. Commun.*, vol. 64, no. 1, pp. 187–200, Jan. 2016.

[7] L. Fan, S. Jin, C. K. Wen, and H. Zhang, “Uplink achievable rate for massive MIMO systems with low-resolution ADC,” *IEEE Commun. Lett.*, vol. 19, no. 12, pp. 2186–2189, Oct. 2015.

[8] H. Xie, F. Gao, and S. Jin, “An overview of low-rank channel estimation for massive MIMO systems,” *IEEE Access*, vol. 4, pp. 7313–7321, 2016.

[9] S. Jin, X. Wang, Z. Li, K.-K. Wong, Y. Huang, and X. Tang, “On massive MIMO zero-forcing transceivers using time-shifted pilots,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 59–74, Jan. 2016.

[10] A. Adhikary, J. Nam, J.-Y. Ahn, and G. Caire, “Joint spatial division and multiplexing—The large-scale array regime,” *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6441–6463, Oct. 2013.

[11] J. Nam, A. Adhikary, J.-Y. Ahn, and G. Caire, “Joint spatial division and multiplexing: Opportunistic beamforming, user grouping and simplified downlink scheduling,” *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 876–890, Oct. 2014.

[12] A. Adhikary et al., “Joint spatial division and multiplexing for mm-Wave channels,” *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1239–1255, Jun. 2014.

[13] C. Sun, X. Q. Gao, S. Jin, M. Matthaiou, Z. Ding, and C. Xiao, “Beam division multiple access transmission for massive MIMO communications,” *IEEE Trans. Commun.*, vol. 63, no. 6, pp. 2170–2184, Jun. 2015.

[14] H. Xie, F. Gao, S. Jin, J. Fang, and Y.-C. Liang, “Channel estimation for TDD/FDD massive MIMO systems with channel covariance computing,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 4206–4218, Jun. 2018.

[15] Y. Han, T.-H. Hsu, C.-K. Wen, K.-K. Wong, and S. Jin, “Efficient downlink channel reconstruction for FDD multi-antenna systems,” *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3161–3176, Jun. 2019.
S. Dahiya and A. K. Sing, “Channel estimation and channel tracking for massive MIMO,” IEEE Trans. Wireless Commun., vol. 17, no. 3, pp. 1907–1922, Mar. 2018.

Y. Zeng and R. Zhang, “Millimeter wave MIMO with lens antenna array: A new path division multiplexing paradigm,” IEEE Trans. Commun., vol. 64, no. 4, pp. 1557–1571, Apr. 2016.

H. Xie, F. Gao, S. Zhang, and S. Jin, “A unified transmission strategy for TDD/FDD massive MIMO systems with spatial basis expansion model,” IEEE Trans. Veh. Technol., vol. 66, no. 4, pp. 3170–3184, Apr. 2017.

S. Daiya and A. K. Sing, “Channel estimation and channel tracking for correlated block-fading channels in massive MIMO systems,” Digital Commun. Netw., vol. 4, no. 2, pp. 138–147, Apr. 2018.

Q. Qin, L. Gui, B. Gong, and S. Luo, “Sparse channel estimation for massive MIMO-OFDM systems over time-varying channels,” IEEE Access, vol. 6, pp. 33740–33751, 2018.

J. Ma, S. Zhang, H. Li, F. Gao, and S. Jin, “Sparse Bayesian learning for the time-varying massive MIMO channels: Acquisition and tracking,” IEEE Trans. Commun., vol. 67, no. 3, pp. 1925–1938, Mar. 2019.

A. M. Sayeed, “Deconstructing multiantenna fading channels,” IEEE Trans. Signal Process., vol. 50, no. 10, pp. 2563–2579, Oct. 2002.

R. Dehghannasiri, M. S. Esfahani, and E. R. Dougherty, “Intrinsically Bayesian robust Kalman filter: An innovation process approach,” IEEE Trans. Signal Process., vol. 65, no. 10, pp. 2531–2546, May 2017.

R. Dehghannasiri, M. S. Esfahani, X. Qian, and E. R. Dougherty, “Optimal Bayesian Kalman filtering with prior update,” IEEE Trans. Signal Process., vol. 66, no. 8, pp. 1982–1996, Apr. 2018.

L. You, X. Q. Gao, A. L. Swindlehurst, and W. Zhong, “Channel acquisition for massive MIMO-OFDM with adjustable phase shift pilots,” IEEE Trans. Signal Process., vol. 64, no. 6, pp. 1461–1476, Mar. 2016.

K. Liu, V. Raghavan, and A. M. Sayeed, “Capacity scaling and spectral efficiency in wide-band correlated MIMO channels,” IEEE Trans. Inf. Theory, vol. 49, no. 10, pp. 2504–2526, Oct. 2003.

J. Zhao, F. Gao, W. Jia, S. Zhang, S. Jin, and H. Lin, “Angle domain hybrid precoding and channel tracking for millimeter wave massive MIMO systems,” IEEE Trans. Wireless Commun., vol. 16, no. 10, pp. 6858–6880, Oct. 2017.

Z. Gao, L. Dai, Z. Wang, and S. Chen, “Spatially common sparsity based adaptive channel estimation and feedback for FDD massive MIMO,” IEEE Trans. Signal Process., vol. 63, no. 23, pp. 6169–6183, Dec. 2015.

Q. Zhang and S. A. Kassam, “Finite-state Markov model for Rayleigh fading channels,” IEEE Trans. Commun., vol. 47, no. 11, pp. 1688–1692, Nov. 1999.

P. Parvazi and A. B. Gershman, “Direction-of-arrival and spatial signature estimation in antenna arrays with pairwise sensor calibration,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., Mar. 2010, pp. 2618–2621.

R. Prasad and C. R. Murthy, “Bayesian learning for joint sparse ODFM channel estimation and data detection,” in Proc. IEEE Global Telecommun. Conf. (GLOBECOM), Dec. 2010, pp. 1–6.

J. A. Ziniel, “Message passing approaches to compressive inference under structured signal priors,” Ph.D. dissertation, Dept. Elect. Comput. Eng., Ohio State Univ., Columbus, OH, USA, 2014.

D. P. Wipf and B. D. Rao, “Sparse Bayesian learning for basis selection,” IEEE Trans. Signal Process., vol. 52, no. 8, pp. 2153–2164, Aug. 2004.

T. Kailath, A. H. Sayed, and B. Hassibi, Linear Estimation. Upper Saddle River, NJ, USA: Prentice-Hall, 2000.

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, Feb. 2001.

C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics). Cambridge U.K.: Springer, 2016.

Muye Li received the B.S. degree in communication engineering from Xidian University, Xi’an, China, in 2013, and the M.E. degree in signal and information processing, the M.E. degree in signal and information processing, from Xidian University, Xi’an, China, in 2013. He is currently with the State Key Laboratory of Integrated Services Networks, Xidian University. His research interests include MIMO-OFDM systems, relay networks, and detection and parameter estimation theory.

Nan Zhao (S’08–M’11–SM’16) received the B.S. degree in electronics and information engineering, the M.E. degree in signal and information processing, and the Ph.D. degree in information and communication engineering from the Harbin Institute of Technology, Harbin, China, in 2005, 2007, and 2011, respectively. He is currently an Associate Professor with the Dalian University of Technology, China. His recent research interests include UAV communications, interference alignment, and physical-layer security.

WeiLe Zhang received the B.S. and Ph.D. degrees in information and communication engineering from Xi’an Jiaotong University, Xi’an, China, in 2006 and 2012, respectively. From 2010 to 2011, he was a Visiting Scholar with the Department of Computer Science, University of California at Santa Barbara, Santa Barbara, CA, USA. He is currently an Associate Professor with the Ministry of Education Key Laboratory for Intelligent Networks and Network Security, Xi’an Jiaotong University. His research interests include broadband wireless communications, MIMO, array signal processing, and localization in wireless networks.

Xianbin Wang (S’98–M’99–SM’06–F’17) received the Ph.D. degree in electrical and computer engineering from the National University of Singapore in 2001. From January 2001 to July 2002, he was a System Designer at STMicroelectronics. He was with the Communications Research Centre Canada (CRC) as a Research Scientist/Senior Research Scientist from July 2002 to December 2007. He is currently a Professor and a Tier 1 Canada Research Chair at Western University, Canada. He has over 380 peer-reviewed journal and conference papers, in addition to 29 granted and pending patents and several standard contributions. His current research interests include 5G technologies, the Internet of Things, communications security, machine learning, and intelligent communications. Dr. Wang is a fellow of the Canadian Academy of Engineering and an IEEE Distinguished Lecturer. He has received many awards and recognitions, including the Canada Research Chair, the CRC Presidents Excellence Award, the Canadian Federal Government Public Service Award, the Ontario Early Researcher Award, and six IEEE Best Paper Awards. He was involved in many IEEE conferences, including GLOBECOM, ICC, VTC, PIMRC, WCNC, and CWIT, in different roles such as the Symposium Chair, a Tutorial Instructor, the Track Chair, the Session Chair, and the TPC Co-Chair. He was also an Associate Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS from 2007 to 2011 and IEEE WIRELESS COMMUNICATIONS LETTERS from 2011 to 2016. He currently serves as an Editor/Associate Editor for IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON BROADCASTING, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.