Exploiting Dynamic Sparsity for Downlink FDD-Massive MIMO Channel Tracking

Lixiang Lian, *Student Member, IEEE*, An Liu, *Senior Member, IEEE*, and Vincent K. N. Lau, *Fellow, IEEE*

***Abstract*—Accurate channel tracking with a small pilot overhead is vital for real-time massive multiple-input and multiple-output (MIMO) communication over a dynamic channel. Recently, compressive sensing has been applied to reduce the pilot overheads by exploiting the spatial and/or temporal correlation of massive MIMO channels. However, most existing channel estimation and tracking algorithms are based on oversimplified channel models with restrictive assumptions, and thus perform poorly under realistic channels. In this paper, we consider downlink frequency-division duplexing-massive MIMO system operating with limited scattering around the base station and flat fading channel is considered. We propose a *two-dimensional Markov Model* (2D-MM) to capture the 2-D dynamic sparsity (i.e., structured sparsity in the spatial domain and probabilistic temporal dependency of channel in the temporal domain) of massive MIMO channels. The 2D-MM has the flexibility to model different propagation environments in practice. We derive an effective message passing algorithm called *dynamic turbo orthogonal approximate message passing* (D-TOAMP) to recursively track a dynamic massive MIMO channel with a 2D-MM prior. The proposed D-TOAMP algorithm does not require knowledge of the 2D-MM channel parameters, which could be automatically learned through the expectation maximization framework. Extensive simulations show that the proposed D-TOAMP can achieve significant gains over the existing algorithms under realistic channels.**

***Index Terms*—Massive MIMO, channel tracking, 2D dynamic sparsity.**

I. Introduction

MASSIVE multiple-input and multiple-output (MIMO) is a promising technology for future wireless systems [1]. It can provide large spatial multiplexing gain as well as array gain to enhance both the capacity and energy efficiency of wireless systems [2]. Accurate channel state information (CSI) estimation and tracking is essential to reap the benefit of massive MIMO for communication over a dynamic wireless channel [3]–[8].

Manuscript received March 6, 2018; revised July 25, 2018 and October 18, 2018; accepted January 22, 2019. Date of publication January 30, 2019; date of current version March 7, 2019. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Antti Tölli. This work was supported in part by the Science and Technology Program of Shenzhen, China, under Grant JCYJ20170818113908577, in part by the National Natural Science Foundation of China under Grant 61571383, and in part by the Research Grants Council under Project 16209916. The work of A. Liu was supported by the China Recruitment Program of Global Young Experts. *(Corresponding author: An Liu.)*

L. Lian and V. K. N. Lau are with the Department of Electronics and Communication Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: llianab@connect.ust.hk; eeknlau@ust.hk).

A. Liu is with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: anliu@zju.edu.cn).

Digital Object Identifier 10.1109/TSP.2019.2896179

In a frequency-division duplexing (FDD) system, if we use conventional channel estimation (CE) methods, such as least squares (LS) and minimum mean square error (MMSE) [9] to estimate the downlink channel at the user side, the number of pilot symbols should be at least the same as the number of antennas M at the base station (BS), which would be prohibitively large for massive MIMO system. In practice, due to the limited number of scatterers in the environment, the massive MIMO channel can be quite sparse under an appropriate spatial basis [10]. Various compressive sensing (CS)-based downlink CE schemes have been developed to exploit the inherit sparsity of massive MIMO channels so as to reduce the pilot training overhead, such as orthogonal matching pursuit (OMP) and compressive sampling matching pursuit (CoSaMP) [11]. Several channel estimation algorithms exploit the spatial and/or temporal correlation of the channel to further reduce the pilot overhead and improve the real-time channel tracking performance, as summarized below.

Algorithms exploiting the spatial correlation[1] of the channel support: In [12], it is shown that the angular domain *channel support* (index set of the nonzero elements in the angular domain channel vector) has a burst structure due to the physical scattering structure in the environment, and this burst sparsity structure has been exploited to design a burst-least absolute shrinkage and selection operator (LASSO) algorithm in [12]. Recently, a structured Turbo CS algorithm with Markov channel prior was proposed in [13] to exploit the clustered sparse structure in the spatial domain for massive MIMO channel estimation.

Algorithms exploiting the temporal correlation of the channel: Since the propagation environment is dynamically changing with the temporal correlation, the previously estimated channel can provide some prior information about the current channel. In [14], the channel is modeled by a Markov process and the classical Kalman filter is used to track the channel. However, the method in [14] cannot exploit the sparsity in massive MIMO channels. In [3]–[5] and [6], both the prior information obtained from the temporal correlation and the channel sparsity have been exploited to design the prior-aided sparse channel tracking schemes.

Algorithms exploiting the frequency (delay)-temporal correlation of the channel support for OFDM systems: In [7] and [8], adaptive channel estimation schemes are designed for broad band systems where orthogonal frequency-division multiplexing (OFDM) is used. They assume that the sparsity

[1]In this paper, spatial correlation refers to the structured spatial sparsity of a massive MIMO channel in which the non-zero elements of the angular domain channel tend to concentrate on a few clusters.

structure is shared by subchannels of different subcarriers (or delay domain channels between the user and different transmit antennas at the BS), and such sparsity is almost unchanged in multiple time blocks.

However, the existing algorithms have the following drawbacks. First, the algorithms exploiting the spatial and/or temporal correlation of the channel support rely on some restrictive assumptions. For example, the burst LASSO algorithm in [12] only works when all bursts of the channel vector have similar sizes, which is not satisfied by practical massive MIMO channels with random burst sizes. [7] and [8] assume that the channel supports across multiple time slots are approximately the same. However, in practice, channel support is time-varying and may undergo a sudden change within two adjacent time slots. Second, many existing algorithms [4], [7] are batch algorithms which require the collection of measurements across multiple time slots to recover a batch of unknown signals simultaneously. Such batch algorithms are offline, slow, and require linearly increasing memory with the sequence length. In practice, it is desirable to design recursive algorithms, which only use the previous signal estimate and current measurements to estimate the current signal, and thus have lower computation complexity and storage requirements. Third, in this paper, we model the fading channel as a dynamic process, and at the same time, maintain spatial sparsity structures as channel evolves over time. Spatial sparsity structure means that the non-zero elements of the angular domain channel tend to concentrate on a few clusters; a dynamic process means the probabilistic dependency of the channels over time. Because these characteristics will cause the sparsity pattern of the angular domain channel spatially and temporally correlated, we call it as *two-dimensional (2D) dynamic sparsity* in the rest of the paper. The existing channel tracking works also exploited the dynamic sparsity. However, [5], [7], [8], [15] failed to model the dynamic evolution of the channel, and just assumed the common sparsity across time or the prior information quality bound; [3], [4], [6], [14], [16]–[18] considered the dynamic evolution of the channel, but they failed to exploit the spatial sparsity structure while the channel evolves. Therefore, there is no existing paper which exploits the structured spatial sparsity and the probabilistic temporal dependency of the channels jointly to track the dynamic channels.

In this paper, we consider downlink FDD-massive MIMO system and propose a systematic framework called *Dynamic Turbo Orthogonal Approximate Message Passing* (D-TOAMP) to recursively track time-varying channels with low pilot overhead and improved performance. The main contributions are summarized below.

- **Realistic Two-dimensional Markov Channel Model:** We propose a new statistical model called the *two-dimensional Markov Model* (2D-MM) to capture the 2D dynamic sparsity of massive MIMO channels. The 2D-MM has the flexibility to model different propagation environments that occur in practice. Moreover, we verify the validity of the 2D-MM using realistic channel models. To the best of our knowledge, this is the first work which uses a 2D-MM to model the 2D dynamic sparsity of sparse massive MIMO channels.
- **Design of Dynamic Turbo-OAMP Algorithm:** By combining the turbo approach and the orthogonal approximate message passing (OAMP) [19]–[21], we propose an efficient sum-product message passing algorithm to recursively track the dynamic channels with a 2D-MM prior. The OAMP in [19]–[21] only works for independent and identically distributed (i.i.d.) priors. We extend the OAMP to D-TOAMP, which works for the 2D-MM prior.

The rest of the paper is organized as follows. In Section II, we present the system model. In Section III, we introduce the 2D-MM channel prior. In Section IV, we present the sparse channel tracking formulation. In Section V, we present the proposed D-TOAMP algorithm and give a complexity analysis. Simulation results and conclusion are given in Section VI and VII, respectively.

Notations: Upper case and lower case bold face letters denote matrices and vectors, respectively. $\mathbf{X}^{-1}$, $\mathbf{X}^T$ and $\mathbf{X}^H$ denote the inverse, transpose and conjugate transpose of matrix $\mathbf{X}$, respectively. $|\Omega|$ denotes the cardinality of a set Ω. $\|\mathbf{x}\|$ means the ℓ_2 norm of vector $\mathbf{x}$. $\mathcal{CN}(x;\mu,\nu)$ denotes the PDF of a complex Gaussian random variable x with mean μ and variance ν.

II. System Model

A. Downlink Training

Consider a massive MIMO system with one BS serving a single antenna user.[2] The BS is equipped with $M \gg 1$ antennas. To estimate the downlink channel $\mathbf{h}_t^H \in \mathbb{C}^{1\times M}$ at time slot t, the BS transmits P pilot sequences $\mathbf{u}_{p,t} \in \mathbb{C}^{M\times 1}, p = 1,\ldots,P$. The received signal $\mathbf{y}_t^H \in \mathbb{C}^{1\times P}$ at time slot t is

$$\mathbf{y}_t^H = \mathbf{h}_t^H \mathbf{U}_t + \mathbf{n}_t^H, \tag{1}$$

where $\mathbf{U}_t = [\mathbf{u}_{1,t},\ldots,\mathbf{u}_{P,t}] \in \mathbb{C}^{M\times P}$ is the pilot matrix, and $\mathbf{n}_t \sim \mathcal{CN}(0,\sigma_e^2\boldsymbol{I})$ is the additive complex Gaussian noise.

In this paper, we consider massive MIMO at the BS side. For large number of antennas, the spatial resolution of the angular basis increases. Hence, under limited scattering environment, the channel will be sparse under the angular basis.

B. Massive MIMO Channel Model

We consider flat fading massive MIMO channel with limited scattering around the BS. For clarity, we focus on the case when the BS is equipped with a *half-wavelength space* uniform linear array (ULA). In this case, the downlink channel vector $\mathbf{h}_t \in \mathbb{C}^M$ can be modeled as [10]

$$\mathbf{h}_t = \sum_{c=1}^{N_c}\sum_{b=1}^{N_b} a_{t,c,b}\mathbf{a}\left(\vartheta_{t,c,b}\right), \tag{2}$$

where N_c stands for the number of scattering clusters, N_b stands for the number of sub-paths per cluster, $a_{t,c,b}$ and $\vartheta_{t,c,b}$ stand

[2]Note that we focus on downlink channel estimation, where the base station sends some common pilots for all users to estimate their own channels. In a multi-user downlink massive MIMO system, each user can independently perform channel tracking using the proposed D-TOAMP based on the received signal from the common pilots. Therefore, without loss of generality, we can focus on the algorithm design for a reference user, and the proposed D-TOAMP can be directly applied to a multi-user massive MIMO system.

for the complex channel gain and the azimuth angle of departure (AoD) corresponding to the b-th sub-path in the c-th scattering cluster at time slot t. The steering vector $\mathbf{a}(\vartheta) \in \mathbb{C}^M$ for ULA is

$$\mathbf{a}(\vartheta) = \left[1, e^{-j\pi\sin(\vartheta)}, \ldots, e^{-j\pi(M-1)\sin(\vartheta)}\right]^T.$$

C. Off-Grid Basis for Massive MIMO Channels

In this subsection, we describe the angular domain channel representation at time slot t. For ease of notation, we drop the time index t. The true AoDs could be denoted as $\{\vartheta_1, \ldots, \vartheta_L\}$ where $L = N_c N_b$. Let $\{\hat{\vartheta}_1, \ldots, \hat{\vartheta}_M\}$ be a uniform sampling grid, which uniformly covers the angular domain $[-\frac{\pi}{2}, \frac{\pi}{2}]$. In practice, the true AoDs usually do not lie exactly on the grid points. To handle the direction mismatch, we adopt an off-grid model. Specifically, if $\vartheta_l \notin \{\hat{\vartheta}_1, \ldots, \hat{\vartheta}_M\}$, and $\hat{\vartheta}_{m_l}$, $m_l \in \{1, \ldots, M\}$ is the nearest grid point to ϑ_l, we write ϑ_l as

$$\vartheta_l = \hat{\vartheta}_{m_l} + \beta_{m_l}, \tag{3}$$

where β_{m_l} is the off-grid gap. Then we have $\mathbf{a}(\vartheta_l) = \mathbf{a}(\hat{\vartheta}_{m_l} + \beta_{m_l})$. The downlink channel $\mathbf{h}$ in (2) has a sparse representation with off-grid basis as given by

$$\mathbf{h} = \mathbf{A}(\boldsymbol{\beta})\,\mathbf{x}, \tag{4}$$

where $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_M]^T$, $\mathbf{A}(\boldsymbol{\beta}) = [\mathbf{a}(\hat{\vartheta}_1 + \beta_1), \ldots, \mathbf{a}(\hat{\vartheta}_M + \beta_M)]$, $\mathbf{x} \in \mathbb{C}^M$ is the sparse angular domain channel, and

$$\beta_{m_l} = \begin{cases} \vartheta_l - \hat{\vartheta}_{m_l}, & l = 1, \cdots L \\ 0, & \text{otherwise} \end{cases}. \tag{5}$$

Note that with the off-grid basis, the model could significantly alleviate the direction mismatch because there always exists some β_{m_l} making (3) hold exactly.

We can also obtain similar sparse representation with off-grid basis for more general 2D antenna arrays. In this case, the steering vector $\mathbf{a}(\vartheta, \varphi)$ can be expressed as a function of the azimuth angle ϑ and elevation angle φ. Please refer to [22] for the detailed expression of $\mathbf{a}(\vartheta, \varphi)$. In this case, the downlink channel vector $\mathbf{h}_t \in \mathbb{C}^M$ can be modeled as [22]

$$\mathbf{h}_t = \sum_{l=1}^{L} a_l \mathbf{a}(\vartheta_l, \varphi_l), \tag{6}$$

where L stands for the total number of sub-paths, $a_l, \vartheta_l, \varphi_l$ stand for the complex channel gain, the azimuth AoD, and elevation AoD corresponding to the l-th sub-path. Then the downlink channel $\mathbf{h}$ in (6) also has a sparse representation with off-grid basis as given by $\mathbf{h} = \mathbf{A}(\boldsymbol{\beta}, \hat{\boldsymbol{\varphi}})\mathbf{x}$, where $\mathbf{A}(\boldsymbol{\beta}, \hat{\boldsymbol{\varphi}}) = [\mathbf{a}(\hat{\vartheta}_1 + \beta_1, \hat{\varphi}_1), \ldots, \mathbf{a}(\hat{\vartheta}_M + \beta_M, \hat{\varphi}_M)]$,

$$\hat{\varphi}_{m_l} = \begin{cases} \varphi_l, & l = 1, \cdots L \\ 0, & \text{otherwise} \end{cases},$$

the definition of m_l can be found in (3). The parameters $(\boldsymbol{\beta}, \hat{\boldsymbol{\varphi}})$ could be learned through the expectation maximization (EM) framework in the proposed D-TOAMP algorithm.

The proposed algorithm in this paper can be applied to general array geometry at the BS with an invertible array response matrix $\mathbf{A}(\mathbf{0}, \mathbf{0})$. For a ULA with half-wavelength inter antenna spacing, the array response matrix $\mathbf{A}$ with $\boldsymbol{\beta} = \mathbf{0}$ is a discrete Fourier transform (DFT) matrix [5], [7], which is invertible. Simulations also show that the array response matrix $\mathbf{A}$ is invertible for many other general array geometries, including randomly placed antenna arrays.

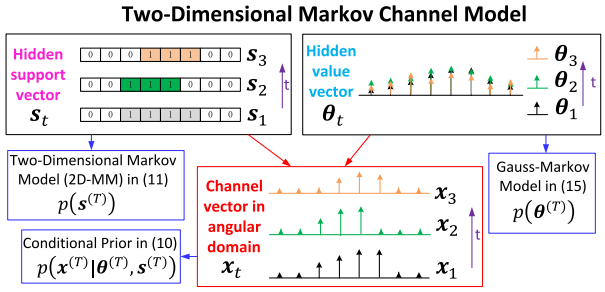


Fig. 1. Two-dimensional Markov channel model.

III. Two-Dimensional Markov Channel Model

Using the angular domain channel representation, the downlink channel at time slot t can be expressed as

$$\mathbf{h}_t = \mathbf{A}(\boldsymbol{\beta}_t)\,\mathbf{x}_t, \tag{7}$$

where $\mathbf{x}_t$ is the angular domain channel vector at time slot t.

The channel model in (7) lacks a probability model for $\mathbf{x}_t$. Such a probability model provides the foundation for exploiting the 2D dynamic sparsity of massive MIMO channels. In existing work, there are some attempts to exploit the sparsity for massive MIMO CE under a very simple assumption for i.i.d. sparsity [11]. However, in practice, due to the clustered scattering, the support of the massive MIMO channels will not be i.i.d. distributed. In [12], a burst sparsity is introduced to account for the clustered scattering. However, this model is deterministic and cannot capture a more complicated clustered scattering structure with random cluster sizes and locations. Furthermore, due to slowly changing propagation environment, the dynamic scattering structures are temporally correlated. Previous estimated channel can provide prior information to enhance the current CE efficiency [3]–[8]; however, the assumptions of the deterministic support structure shared by consecutive time slots in [7], [8] and the prior information quality bound in [5] are too restrictive.

> Challenge 1: Propose a probabilistic channel model to capture a more realistic 2D dynamic sparsity of the massive MIMO channels.

In this paper, we introduce a 2D Markov model to capture a more realistic 2D dynamic sparsity of the massive MIMO channels. Fig. 1 illustrates the high level structure of the 2D-MM for the massive MIMO channels $\mathbf{x}_1, \ldots, \mathbf{x}_T$.

Let Ω_t denote the index set of non-zero elements of $\mathbf{x}_t$, which is called the *channel support* at time slot t. In order to characterize the 2D dynamic sparsity of a dynamic channel $\mathbf{x}^{(T)} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$, we adopt a probabilistic signal model with two hidden random processes $\mathbf{s}^{(T)} =$

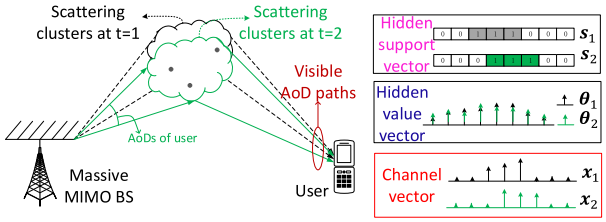


Fig. 2. Illustration of the 2D dynamic sparsity of the massive MIMO channel for $T = 2$. Due to limited and clustered scattering at the BS, the hidden support vector $\mathbf{s}_t$ will be sparse with clustered non-zero elements. Due to the slowly changing scattering environment, the hidden support vector $\mathbf{s}_t$ and hidden value vector $\boldsymbol{\theta}_t$ will be temporally correlated.

$\{\mathbf{s}_1, \ldots, \mathbf{s}_T\}$ and $\boldsymbol{\theta}^{(T)} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_T\}$. The binary vector $\mathbf{s}_t = [s_{t,1}, \ldots, s_{t,M}]^T \in \{0,1\}^M$, with $s_{t,m} = 1$ if $m \in \Omega_t$, and $s_{t,m} = 0$ otherwise, represents the *hidden support vector* at time slot t, which describes the 2D dynamic sparsity of the channel sparsity pattern. The complex-valued vector $\boldsymbol{\theta}_t = [\theta_{t,1}, \ldots, \theta_{t,M}]^T \in \mathbb{C}^M$ with $\theta_{t,m} = x_{t,m}$ if $s_{t,m} = 1$ represents the *hidden value vector* at time slot t, which characterizes the temporal correlation of the channel coefficients. The dynamic channel can be modeled as

$$x_{t,m} = s_{t,m} \cdot \theta_{t,m}, \; t = 1, \ldots, T, 1 \le m \le M, \tag{8}$$

where $s_{t,m}$ denotes whether there is an active path from the m-th AoD direction in the t-th time slot at the BS, and $\theta_{t,m}$ denotes the corresponding complex path gain. Then the 2D-MM channel prior distribution (joint distribution of $\mathbf{x}^{(T)}, \mathbf{s}^{(T)}, \boldsymbol{\theta}^{(T)}$) is given by

$$\begin{aligned} &p\left(\mathbf{x}^{(T)}, \mathbf{s}^{(T)}, \boldsymbol{\theta}^{(T)}\right) \\ &= \underbrace{p\left(\mathbf{x}^{(T)} \middle| \mathbf{s}^{(T)}, \boldsymbol{\theta}^{(T)}\right)}_{\text{channel vector}} \underbrace{p\left(\mathbf{s}^{(T)}\right)}_{\text{hidden support}} \underbrace{p\left(\boldsymbol{\theta}^{(T)}\right)}_{\text{hidden value}}, \end{aligned} \tag{9}$$

where the probability model for channel vectors $\mathbf{x}^{(T)}$ is conditioned on the hidden support vectors and hidden value vectors, the hidden support vectors $\mathbf{s}^{(T)}$ form a 2D-MM, and the hidden value vectors $\boldsymbol{\theta}^{(T)}$ form a Gauss-Markov process, as detailed below.

A. Probability Model for Channel Vector

The conditional prior $p(\mathbf{x}^{(T)}|\mathbf{s}^{(T)}, \boldsymbol{\theta}^{(T)})$ is given by

$$\begin{aligned} p\left(\mathbf{x}^{(T)} \middle| \mathbf{s}^{(T)}, \boldsymbol{\theta}^{(T)}\right) &= \prod_{t=1}^{T} \prod_{m=1}^{M} \underbrace{p\left(x_{t,m} | s_{t,m}, \theta_{t,m}\right)}_{f_{t,m}\left(x_{t,m}, s_{t,m}, \theta_{t,m}\right)} \\ &= \prod_{t=1}^{T} \prod_{m=1}^{M} \delta\left(x_{t,m} - \theta_{t,m} s_{t,m}\right), \end{aligned} \tag{10}$$

where $\delta(\cdot)$ is the Dirac delta function. Conditioned on $\mathbf{s}^{(T)}, \boldsymbol{\theta}^{(T)}$, $x_{t,m}$'s are independent. By definition, $s_{t,m} = 0$ sets $x_{t,m} = 0$, while $s_{t,m} = 1$ sets $x_{t,m} = \theta_{t,m}$.

The 2D-MM channel is motivated by the physical propagation mechanism of radio waves. Physically, each Tx scatterer at the BS side corresponds to an AoD, as shown in Fig. 2. If the m-th AoD path is visible to the user (i.e., the signal of the m-th AoD path reflected/scattered from the Tx scatterer can reach the user), the m-th element of angular domain channel vector $\mathbf{x}_t$ will be non-zero. In practice, there are limited Tx scatterers at the BS, i.e., the BS is elevated high or the carrier frequency is high, so only part of the AoD paths can reach the user. The hidden support vector $\mathbf{s}_t$ represents the AoD paths that can be seen by the user. Moreover, since we consider a channel tracking problem where an invisible AoD path at the current time slot may become visible at the next time slot (and vice-versa), it is necessary to use a hidden value vector $\boldsymbol{\theta}_t$ to model the path gains of all the M potential AoD paths, as shown by [3] and [4]. Therefore, it is natural to model the actual channel vector $\mathbf{x}_t$ as the product of $\mathbf{s}_t$ and $\boldsymbol{\theta}_t$.

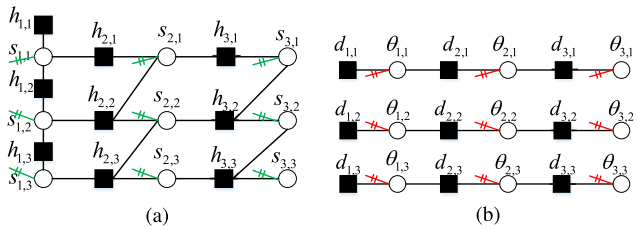


Fig. 3. Factor graphs of hidden support and value vectors when $M = 3$ and $T = 3$. (a) Factor graph of the 2D-MM of the hidden support vectors; (b) factor graph of Gauss-Markov model of the hidden value vectors.

B. Two-Dimensional Markov Model of Hidden Support Vector

Due to the clustered structure of the scatterers at the BS side, the non-zero elements in $\mathbf{s}_t$ will concentrate on a few clusters [12], where each cluster corresponds to a transmit scattering cluster. This spatially clustered structure implies that $s_{t,m}$ depends on $s_{t,m-1}$, e.g., if $s_{t,m-1} = 1$, then there is a higher probability that $s_{t,m}$ is also 1. Moreover, it has been verified in [3] and [4] that the channel supports often change slowly over time, which implies that $s_{t,m}$ also depends on $s_{t-1,m}$, e.g., if $s_{t-1,m} = 1$, then there is a higher probability that $s_{t,m}$ is also 1. Such 2D dynamic sparsity of hidden support vectors can be naturally modeled as the following 2D-MM [23]:

$$\begin{aligned} p\left(\mathbf{s}^{(T)}\right) &= \underbrace{p\left(s_{1,1}\right)}_{h_{1,1}\left(s_{1,1}\right)} \prod_{m=2}^{M} \underbrace{p\left(s_{1,m} | s_{1,m-1}\right)}_{h_{1,m}\left(s_{1,m}, s_{1,m-1}\right)} \\ &\times \prod_{t=2}^{T} \left(\underbrace{p\left(s_{t,1} | s_{t-1,1}\right)}_{h_{t,1}\left(s_{t,1}, s_{t-1,1}\right)} \prod_{m=2}^{M} \underbrace{p\left(s_{t,m} | s_{t,m-1}, s_{t-1,m}\right)}_{h_{t,m}\left(s_{t,m}, s_{t,m-1}, s_{t-1,m}\right)} \right), \end{aligned} \tag{11}$$

whose 2D transition probabilities are defined as $\rho_{01}^S = p(s_{1,m} = 1|s_{1,m-1} = 0)$, $\rho_{10}^S = p(s_{1,m} = 0|s_{1,m-1} = 1)$, $\rho_{01}^T = p(s_{t,1} = 1|s_{t-1,1} = 0)$, $\rho_{10}^T = p(s_{t,1} = 0|s_{t-1,1} = 1)$ and $\rho_{bca} = p(s_{t,m} = a|s_{t,m-1} = b, s_{t-1,m} = c)$, where $a, b, c \in \{0,1\}$, $t > 1$ and $m > 1$. The factor graph of the 2D-MM hidden support vector is illustrated in Fig. 3a, where the factor nodes $h_{t,m}$ are the conditional priors (priors) in (11). Note that we can always find a set of transition probabilities $\{\rho_{ba}^S, \rho_{ca}^T, \rho_{bca}\}$ to make the 2D-MM operate in a steady-state [23], such that $p(s_{t,m} = 1) = \lambda, \forall t, m$, where $\lambda > 0$ is the sparsity ratio.

Depending on how $\{\rho_{bca}\}$, $a, b, c \in \{0, 1\}$ are choosing, the prior distribution in (11) can favor static or substantially changing support in the temporal domain. For example, higher ρ_{b11} and smaller $\rho_{b01}, b \in \{0, 1\}$ lead to temporally highly correlated priors of $\mathbf{s}^{(T)}$. Meanwhile, the prior distribution in (11) can characterize the clustered structure of $\mathbf{s}_t$ in the spatial domain. For example, a higher $\rho_{1c1}, c \in \{0, 1\}$ leads to a larger average cluster size, and a smaller $\rho_{0c1}, c \in \{0, 1\}$ leads to a larger average gap between clusters in $\mathbf{s}_t$. As such, the 2D-MM in (11) is a general flexible model to characterize the 2D dynamic sparsity of $\mathbf{s}^{(T)}$.

C. Gauss-Markov Model of Hidden Value Vector

It has been shown in [3] and [4] that the path gains evolve smoothly over time. We can use the spatially independent steady-state Gauss-Markov processes to model the temporal evolution of the hidden value vector as follows [18]:

$$\theta_{t,m} = (1-\alpha)(\theta_{t-1,m} - \zeta) + \alpha w_{t,m} + \zeta, \tag{12}$$

where $\alpha \in [0, 1]$ controls the temporal correlation, $\zeta \in \mathbb{C}$ is the mean of the process, and $w_{t,m} \sim \mathcal{CN}(0, \kappa)$ is the i.i.d Gaussian perturbation with mean 0 and variance κ. Specifically, if $\alpha = 0$, then $\theta_{t,m} = \theta_{t-1,m}$, which means the hidden value vector $\boldsymbol{\theta}_t$ is unchanged over time. If $\alpha = 1$, then $\theta_{t,m} = w_{t,m} + \zeta \sim \mathcal{CN}(\zeta, \kappa)$, which means the hidden value vector $\boldsymbol{\theta}_t$ is i.i.d Gaussian distributed over time. If $0 < \alpha < 1$, based on (12), the conditional probability of $\theta_{t,m}$ could be given by

$$p(\theta_{t,m}|\theta_{t-1,m}) \sim \mathcal{CN}\left(\theta_{t,m}; (1-\alpha)\theta_{t-1,m} + \alpha\zeta, \alpha^2\kappa\right). \tag{13}$$

In the steady-state

$$\theta_{t,m} \sim \mathcal{CN}\left(\zeta, \sigma^2\right), \forall t, m, \tag{14}$$

where the steady-state variance is $\sigma^2 = \frac{\alpha\kappa}{2-\alpha}$. The joint distribution of $\boldsymbol{\theta}^{(T)}$ can be calculated as follows:

$$p\left(\boldsymbol{\theta}^{(T)}\right) = \prod_{m=1}^{M} \underbrace{p(\theta_{1,m})}_{d_{1,m}(\theta_{1,m})} \prod_{t=2}^{T} \underbrace{p(\theta_{t,m}|\theta_{t-1,m})}_{d_{t,m}(\theta_{t,m},\theta_{t-1,m})}. \tag{15}$$

The factor graph of the Markovian hidden value vector is illustrated in Fig. 3b, where the factor nodes $d_{t,m}$ are the conditional priors (priors) in (13).

Finally, the overall factor graph of the 2D-MM channel is illustrated in Fig. 4, where the factor node $f_{t,m}$ is the conditional prior in (10).

D. Verification of Two-Dimensional Markov Channel Model

Compared to other static channel models [7], [8], [12], the proposed 2D-MM provides more flexibility to model more realistic channels. Specifically, in our model, the average cluster size and cluster number in the spatial domain, and the dependency of the channel across time are determined by a set of parameters $\boldsymbol{\rho} \triangleq \{\rho_{ba}^S, \rho_{ca}^T, \rho_{bca}, \alpha, \zeta, \kappa, \sigma^2\}$, $a, b, c \in \{0, 1\}$. For given parameters, the channel realizations could have different spatial-temporal properties, i.e., different cluster numbers and cluster sizes, and different channel evolutions over time. As such, the proposed 2D-MM channel can be used to model various channel realizations in practice, and thus works well for realistic channels. Moreover, the statistic parameters in our model $\boldsymbol{\rho}$ could be automatically learned by the proposed D-TOAMP algorithm based on the EM framework during the recovery process.

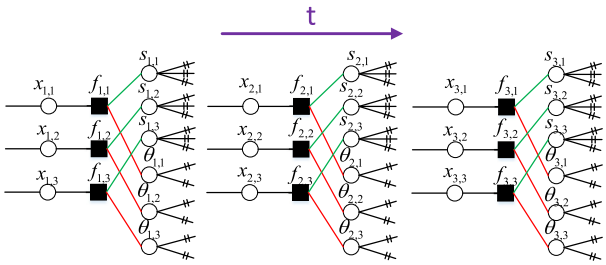


Fig. 4. Factor graph of the 2D-MM channel when $M = 3$ and $T = 3$. The detailed factor graphs for hidden support vector $\mathbf{s}_t$ and hidden value vector $\boldsymbol{\theta}_t$ are illustrated in Fig. 3.

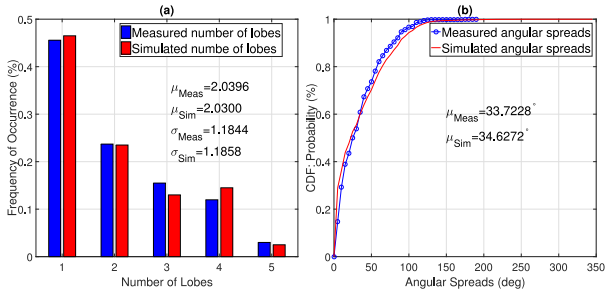


Fig. 5. Comparison of the measured channel property extracted from the 28-GHz mm-SSCM [24] and the simulated channel property extracted from the 2D-MM channel. We set $M = 256$ and $T = 50$. The results are calculated through 200 channel series realizations. (a) Number of AoD SLs; (b) AoD global AS.

In this subsection, we will provide some verifications of the proposed 2D-MM channel. Fig. 5a shows a typical empirical histogram plot of the number of AoD spatial lobes (SLs) extracted from the 28-GHz millimeter-wave statistical spatial channel model (mm-SSCM) proposed in [24], next to the simulated histograms extracted from the proposed 2D-MM channel. It can be seen that the proposed 2D-MM channel prior yields good agreement with the practical mmWave channels.

AoD global angular spread (AS) describes the degree of angular dispersion at the BS over the entire 2π azimuth plane. The AoD global AS in Fig. 5b are computed based on a -10 dB lobe threshold from the measured data in the 28-GHz mm-SSCM [24], and are compared to the simulated values using the 2D-MM channel prior. It can be seen that the statistics of the simulated and measured global AS match well.

IV. Massive MIMO Channel Tracking With 2D Dynamic Sparsity

Using the angular domain channel representation, (1) can be rewritten as a standard CS model as in [5], [12]

$$\mathbf{y}_t = \mathbf{F}_t(\boldsymbol{\beta}_t)\mathbf{x}_t + \mathbf{n}_t, \forall t, \tag{16}$$

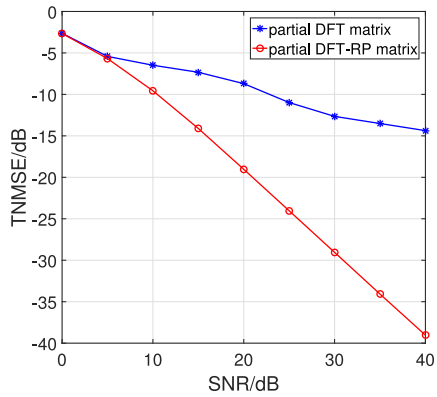


Fig. 6. TNMSE performance (defined in (61)) of the proposed D-TOAMP algorithm versus SNR for different partial orthogonal measurement matrices. Set $T = 50, P = 50, M = 128$, $\lambda = 0.25$, $\rho_{01}^S = 0.025$, $\rho_{10}^S = 0.075$, $\rho_{01}^T = 0.05$, $\rho_{10}^T = 0.05$, $\rho_{111} = 0.9958$, $\rho_{001} = 0.0013$, $\rho_{011} = 0.3276$, $\rho_{101} = 0.3936$, $\kappa = 1, \sigma^2 = \frac{1}{3}, \zeta = 0, \alpha = 0.5$. For partial DFT matrix, $\mathbf{F}_t = \mathbf{S}_t\mathbf{D}$; for partial DFT-RP matrix, $\mathbf{F}_t = \mathbf{S}_t\mathbf{D}\mathbf{R}_t$.

where the measurement matrix $\mathbf{F}_t(\boldsymbol{\beta}_t)$ is given by

$$\mathbf{F}_t(\boldsymbol{\beta}_t) = \mathbf{U}_t^H \mathbf{A}(\boldsymbol{\beta}_t) \in \mathbb{C}^{P\times M}. \tag{17}$$

It is well known that the choice of measurement matrix will affect the CE performance significantly. It is shown in [13], [19] and [20] that a partial orthogonal measurement matrix achieves better performance than an i.i.d. Gaussian matrix under the OAMP algorithm. In order to reduce the signaling overhead between the BS and user, the pilot matrix is designed assuming a fixed off-grid parameter $\boldsymbol{\beta}_t = \mathbf{0}$, i.e., $\mathbf{U}_t^H = \mathbf{S}_t\mathbf{D}\mathbf{R}_t\mathbf{A}(\mathbf{0})^{-1}$, such that $\mathbf{F}_t(\mathbf{0}) = \mathbf{S}_t\mathbf{D}\mathbf{R}_t$, which is referred to as a partial DFT-random permutation (RP) measurement matrix. $\mathbf{S}_t \in \{0,1\}^{P\times M}$ is a selection matrix consisting of randomly selected and reordered P rows of an $M \times M$ identity matrix, $\mathbf{D} \in \mathbb{C}^{M\times M}$ is the DFT matrix, and $\mathbf{R}_t \in \{0,1\}^{M\times M}$ is a RP matrix generated by a randomly reordered $M \times M$ identity matrix. For large antenna arrays, M is large and the measurement matrix $\mathbf{F}_t(\boldsymbol{\beta}_t) = \mathbf{S}_t\mathbf{D}\mathbf{R}_t\mathbf{A}(\mathbf{0})^{-1}\mathbf{A}(\boldsymbol{\beta}_t) \approx \mathbf{S}_t\mathbf{D}\mathbf{R}_t$ is approximately partial orthogonal. Simulations in Section VI verify that such choice of pilot matrix can indeed achieve a good performance.[3]

Remark 1: In the original OAMP algorithm, the measurement matrix is chosen as partial DFT matrix and the sparse signal follows i.i.d distribution. However, as shown in Fig. 6, we find that partial DFT matrix does not work well for signals with complicated correlations, i.e., 2D-MM priors. On the other hand, the partial DFT-RP matrix decorrelates the sparse signal by introducing a RP. Therefore, partial DFT-RP achieves better performance compared to the partial DFT measurement matrix for the D-TOAMP algorithm with non-i.i.d sparse signals.

Based on the observation model in (16) and the 2D-MM for the massive MIMO channels $\mathbf{x}^{(t)}$ in (9), our primary goal is to recursively track the time-varying channel vector $\mathbf{x}_t$ and optimize the off-grid parameter $\boldsymbol{\beta}_t$ at the t-th time slot, given the observations up to t time slots $\mathbf{y}^{(t)} = \{\mathbf{y}_\tau\}_{\tau=1}^t$ and the (approximately) optimal off-grid parameters up to $(t-1)$ time slot $\boldsymbol{\beta}^{*(t-1)} = \{\boldsymbol{\beta}_\tau^*\}_{\tau=1}^{t-1}$ in (16). Because $\boldsymbol{\beta}^{*(t-1)}$ is fixed for the channel tracking problem at the t-th time slot, we omit $\boldsymbol{\beta}^{*(t-1)}$ in the probability expressions for simplicity. In particular, for given $\boldsymbol{\beta}_t$, we are interested in computing the marginal posterior $p(x_{t,m}|\mathbf{y}^{(t)}, \boldsymbol{\beta}_t)$, where

$$\begin{aligned} &p\left(x_{t,m}|\mathbf{y}^{(t)}, \boldsymbol{\beta}_t\right) \\ &\propto \sum_{\mathbf{s}^{(t)}} \int_{\mathbf{x}_{-(t,m)}^{(t)}, \boldsymbol{\theta}^{(t)}} p\left(\mathbf{x}^{(t)}, \mathbf{s}^{(t)}, \boldsymbol{\theta}^{(t)}, \mathbf{y}^{(t)}|\boldsymbol{\beta}_t\right) \\ &= \sum_{\mathbf{s}^{(t)}} \int_{\mathbf{x}_{-(t,m)}^{(t)}, \boldsymbol{\theta}^{(t)}} p\left(\mathbf{s}^{(t)}\right) p\left(\boldsymbol{\theta}^{(t)}\right) \prod_{\tau,m} p\left(x_{\tau,m}|s_{\tau,m}, \theta_{\tau,m}\right). \\ &\quad \times \prod_{i=1}^{P}\prod_{\tau=1}^{t-1} p\left(y_{\tau,i}|\mathbf{x}_\tau\right) p\left(y_{t,i}|\mathbf{x}_t, \boldsymbol{\beta}_t\right). \end{aligned} \tag{18}$$

$\mathbf{x}_{-(t,m)}^{(t)}$ denotes the vector collections $\{\mathbf{x}_\tau\}_{\tau=1}^t$ excluding the element $x_{t,m}$, $p(y_{t,i}|\mathbf{x}_t, \boldsymbol{\beta}_t) = \mathcal{CN}(y_{t,i}; \mathbf{f}_{t,i}\mathbf{x}_t, \sigma_e^2)$, and $y_{t,i}$ is the i-th element of $\mathbf{y}_t$, $\mathbf{f}_{t,i}$ is the i-th row of $\mathbf{F}_t(\boldsymbol{\beta}_t)$. We use $\propto$ to denote equality after scaling. On the other hand, the optimal off-grid parameter $\boldsymbol{\beta}_t$ is obtained by maximum likelihood (ML) as follows:

$$\begin{aligned} \boldsymbol{\beta}_t^\star &= \arg\max_{\boldsymbol{\beta}_t} \ln p\left(\mathbf{y}^{(t)}, \boldsymbol{\beta}_t\right) \\ &= \arg\max_{\boldsymbol{\beta}_t} \ln \int_{\mathbf{x}^{(t)}} p\left(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}, \boldsymbol{\beta}_t\right) d\mathbf{x}^{(t)}. \end{aligned} \tag{19}$$

Once we obtain the ML estimate of $\boldsymbol{\beta}_t^\star$, and the associated conditional marginal posterior $p(x_{t,m}|\mathbf{y}^{(t)}, \boldsymbol{\beta}_t^\star)$, we can obtain the MMSE estimates of $\{x_{t,m}\}$, $\hat{x}_{t,m} = \mathrm{E}(x_{t,m}|\mathbf{y}^{(t)}, \boldsymbol{\beta}_t^\star)$, where the expectation is over the marginal posterior $p(x_{t,m}|\mathbf{y}^{(t)}, \boldsymbol{\beta}_t^\star)$.

It is very challenging to calculate the exact posterior in (18) because the factor graph of the underlying model has loops. In the next section, we propose a D-TOAMP algorithm to approximately calculate the marginal posteriors $\{p(x_{t,m}|\mathbf{y}^{(t)}, \boldsymbol{\beta}_t)\}$ and the associated MMSE estimates, and use an in-exact majorization-minimization (MM) method (which is a generalization of the EM method) [22] to find an approximate solution of (19). One major approximation in the proposed algorithm is that the approximate message passing algorithm (D-TOAMP) may not find the exact posterior in (18) due to the presence of loops in the associated factor graph. Assume for fixed $\boldsymbol{\beta}_t$, D-TOAMP can find the exact posterior in (18), then we can show that the proposed algorithm will converge to a stationary point of the ML problem in (19) for $\boldsymbol{\beta}_t$. The state evolution analysis in [19]–[21] implies that the approximate posterior of $\mathbf{x}_t$ obtained by the OAMP-based algorithm can be quire accurate. As such, the proposed algorithm is expected to achieve a good performance. Indeed, the proposed D-TOAMP algorithm is shown in the simulations to have significant gain over various state-of-the-art baselines.

Challenge 2: There is no closed-form posterior distribution in (18) and it is difficult to obtain the closed-form expression of $\boldsymbol{\beta}_t^\star$ in (19).

[3]Note that the pilot sequence in (17) can be static or time varying. In both cases, they achieve similar performances.

V. Dynamic Turbo-OAMP Algorithm

The basic idea of the D-TOAMP algorithm is to simultaneously approximate the marginal posterior $p(x_{t,m}|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$ exploiting the 2D-MM prior and maximize the log-likelihood $\ln p(\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$ with respect to $\boldsymbol{\beta}_t$ as in (19) at the t-th time slot. In summary, at the t-th time slot, the D-TOAMP algorithm (Algorithm 1) performs iterations between the following two major steps until convergence.

- **D-TOAMP-E Step:** Given $\boldsymbol{\beta}_t$, calculate the approximate marginal posterior $\hat{p}(x_{t,m}|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$ by combining the OAMP and 2D-MM prior via the turbo framework, as elaborated in Section V-B. Then $p(\mathbf{x}_t|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$ can be approximated by $\hat{p}(\mathbf{x}_t|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)=\prod_m \hat{p}(x_{t,m}|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$.
- **D-TOAMP-M Step:** Given $p(\mathbf{x}_t|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)\approx\hat{p}(\mathbf{x}_t|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$, construct a surrogate function (lower bound) for the objective function $\ln p(\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$, then maximize the surrogate function with respect to $\boldsymbol{\beta}_t$, as elaborated in Section V-A.

In the following, we first elaborate the D-TOAMP-M step, which is an extension of the in-exact MM method in [22]. Because the surrogate function in D-TOAMP-M step requires the calculation of the posterior $p(\mathbf{x}_t|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$, we elaborate how to approximately calculate the posterior $p(\mathbf{x}_t|\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$ in the D-TOAMP-E step.

A. D-TOAMP-M Step (In-Exact MM)

It is difficult to directly maximize the log-likelihood function $\ln p(\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$, because there is no closed-form expression due to the multi-dimensional integration over $\mathbf{x}^{(t)}$ as in (19). To make the problem tractable, in the D-TOAMP-M Step, we maximize a surrogate function of $\ln p(\mathbf{y}^{(t)},\boldsymbol{\beta}_t)$ with respect to $\boldsymbol{\beta}_t$. Specifically, let $u(\boldsymbol{\beta}_t;\dot{\boldsymbol{\beta}}_t)$ be the surrogate function constructed at some fixed point $\dot{\boldsymbol{\beta}}_t$,which satisfies the following properties:

$$u(\boldsymbol{\beta}_t;\dot{\boldsymbol{\beta}}_t)\le \ln p\left(\mathbf{y}^{(t)},\boldsymbol{\beta}_t\right),\forall\boldsymbol{\beta}_t, \tag{20}$$

$$u(\dot{\boldsymbol{\beta}}_t;\dot{\boldsymbol{\beta}}_t)= \ln p\left(\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t\right), \tag{21}$$

$$\left.\frac{\partial u(\boldsymbol{\beta}_t;\dot{\boldsymbol{\beta}}_t)}{\partial\boldsymbol{\beta}_t}\right|_{\boldsymbol{\beta}_t=\dot{\boldsymbol{\beta}}_t}=\left.\frac{\partial \ln p\left(\mathbf{y}^{(t)},\boldsymbol{\beta}_t\right)}{\partial\boldsymbol{\beta}_t}\right|_{\boldsymbol{\beta}_t=\dot{\boldsymbol{\beta}}_t}. \tag{22}$$

Inspired by the EM algorithm [25], we use the following surrogate function:

$$u(\boldsymbol{\beta}_t;\dot{\boldsymbol{\beta}}_t)=\int p\left(\mathbf{x}_t|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t\right)\ln\frac{p\left(\mathbf{x}_t,\mathbf{y}^{(t)},\boldsymbol{\beta}_t\right)}{p\left(\mathbf{x}_t|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t\right)}d\mathbf{x}_t. \tag{23}$$

It can be shown that the surrogate function in (23) satisfies (20)–(22). Then in the D-TOAMP-M step of the i-th iteration, we update $\boldsymbol{\beta}_t$ as

$$\boldsymbol{\beta}_t^{i+1}=\arg\max_{\boldsymbol{\beta}_t} u(\boldsymbol{\beta}_t;\boldsymbol{\beta}_t^i), \tag{24}$$

where $\boldsymbol{\beta}_t^i$ stands for the value of $\boldsymbol{\beta}_t$ at the i-th iteration. However, the maximization problem in (24) is non-convex and it is difficult to find its optimal solution. We use an in-exact MM method, where $\boldsymbol{\beta}_t^{i+1}$ is obtained by applying gradient update as follows:

$$\boldsymbol{\beta}_t^{i+1}=\boldsymbol{\beta}_t^i+\Delta^i\cdot\left.\frac{\partial u(\boldsymbol{\beta}_t;\boldsymbol{\beta}_t^i)}{\partial\boldsymbol{\beta}_t}\right|_{\boldsymbol{\beta}_t=\boldsymbol{\beta}_t^i}, \tag{25}$$

where Δ^i is the stepsize, which can be determined by backtracking line search [26]. Alternatively, we may use a fixed stepsize as mentioned in [22] to reduce the computational complexity due to backtracking line search.

Algorithm 1: Dynamic Turbo-OAMP Algorithm.

Input: $\{\mathbf{y}_1,\ldots,\mathbf{y}_T\}$, measurement matrix $\mathbf{F}_t(\mathbf{0})$, $\forall t$, and noise variance σ_e^2.
Output: $\{\hat{\mathbf{x}}_1,\ldots,\hat{\mathbf{x}}_T\}$.
for $t=1,\ldots,T$ **do**
  Initialize: $\mathbf{x}_{A,t}^{pri}=\mathbf{0}$, $v_{A,t}^{pri}=\lambda\sigma^2$, $\boldsymbol{\beta}_t^1=\mathbf{0}$, $\forall t$, $i=1$.
  while not converge **do**
    %D-TOAMP-E Step (for given $\boldsymbol{\beta}_t=\boldsymbol{\beta}_t^i$):
    %Module A: LMMSE estimator
    1: Update $\mathbf{x}_{A,t}^{post}$ and $v_{A,t}^{post}$ using (38) and (39).
    2: Update $\mathbf{x}_{B,t}^{pri}=\mathbf{x}_{A,t}^{ext}$ and $v_{B,t}^{pri}=v_{A,t}^{ext}$ using (41) and (42).
    %Module B: 2D-MM-MMSE estimator (Message Passing over $\mathcal{G}_t$)
    3: Message passing over the path $\theta_{t,m}\rightarrow f_{t,m}$ using (44), with $\nu_{d_{t,m}\rightarrow\theta_{t,m}}$ as the input.
    4: Message passing over the path $x_{t,m}\rightarrow f_{t,m}\rightarrow s_{t,m}$ using (46) and (47), with $\mathbf{x}_{B,t}^{pri}$ and $v_{B,t}^{pri}$ as the input.
    5: Hidden support vector estimation in $\mathcal{G}_{s,t}$ through Algorithm 2 in Section V-B3, with $\nu_{s_{t-1,m}\rightarrow h_{t,m}}$ as the input.
    6: Message passing over the path $s_{t,m}\rightarrow f_{t,m}\rightarrow x_{t,m}$ using (49) and (50).
    7: Calculate the posterior distributions $p(x_{t,m}|\mathbf{x}_B^{pri(t)})$ using (51), and update $\mathbf{x}_{B,t}^{post}$ and $v_{B,t}^{post}$ using (52) and (53).
    8: Update $\mathbf{x}_{A,t}^{pri}=\mathbf{x}_{B,t}^{ext}$ and $v_{A,t}^{pri}=v_{B,t}^{ext}$ using (54) and (55).
    9: Repeat Module A and Module B until convergence.
    10: Then $\hat{p}(x_{t,m}|\mathbf{y}^{(t)},\boldsymbol{\beta}_t^i)=\mathcal{CN}(x_{B,t,m}^{post},v_{B,t}^{post})$.
    Output $\hat{p}(\mathbf{x}_t|\mathbf{y}^{(t)},\boldsymbol{\beta}_t^i s)=\mathcal{CN}(\mathbf{x}_{B,t}^{post},v_{B,t}^{post}\boldsymbol{I})$.
    %D-TOAMP-M Step:
    11: Construct the surrogate function $\hat{u}(\boldsymbol{\beta}_t;\boldsymbol{\beta}_t^i)$ in (26) using the approximate posterior output of D-TOAMP-E step, i.e., $\hat{p}(\mathbf{x}_t|\mathbf{y}^{(t)},\boldsymbol{\beta}_t^i)$. Then update the off-grid parameter $\boldsymbol{\beta}_t^{i+1}$ as in (25).
    12: $i=i+1$.
  end while
  13: Output $\hat{\mathbf{x}}_t=\mathbf{x}_{B,t}^{post}$.
  14: Update messages passed to the $(t+1)$-th time slot $\nu_{s_{t,m}\rightarrow h_{t+1,m}}(s_{t,m})$ and $\nu_{d_{t+1,m}\rightarrow\theta_{t+1,m}}(\theta_{t+1,m})$ using (56) and (60).
end for

Based on the convergence proof of the EM algorithm in [27], we can prove that the in-exact MM converges to a stationary solution of the optimization problem (19).

Lemma 1 (Convergence of In-exact MM): Suppose the surrogate function $u(\boldsymbol{\beta}_t;\boldsymbol{\beta}_t^i)$ satisfies (20)–(22). If at each iteration, we do in-exact (gradient) update as in (25) for off-grid parameter $\boldsymbol{\beta}_t$, the iterates generated by the D-TOAMP algorithm converge to a stationary point of Problem (19).

Therefore, if we can calculate the exact posterior $p(\mathbf{x}_t|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t)$ for given $\dot{\boldsymbol{\beta}}_t$, we can construct the surrogate function in (23) and the corresponding D-TOAMP algorithm converges to a stationary point of (19). Unfortunately, in our case, the exact posterior is intractable due to the loops in the factor graph. Thus, in the D-TOAMP-E step, we incorporate the 2D-MM channel prior into the OAMP algorithm to find an approximation of the marginal posterior $p(x_{t,m}|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t)$, i.e., $\hat{p}(x_{t,m}|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t)$, for any given $\dot{\boldsymbol{\beta}}_t$. Then the posterior $p(\mathbf{x}_t|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t)$ can be approximated by $\hat{p}(\mathbf{x}_t|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t)=\prod_m \hat{p}(x_{t,m}|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t)$. Based on the posterior approximation $\hat{p}(\mathbf{x}_t|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t)$, we can construct a tractable surrogate function as

$$\hat{u}(\boldsymbol{\beta}_t;\dot{\boldsymbol{\beta}}_t)=\int \hat{p}\left(\mathbf{x}_t|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t\right)\ln\frac{p\left(\mathbf{x}_t,\mathbf{y}^{(t)},\boldsymbol{\beta}_t\right)}{\hat{p}\left(\mathbf{x}_t|\mathbf{y}^{(t)},\dot{\boldsymbol{\beta}}_t\right)}d\mathbf{x}_t, \quad (26)$$

which is expected to approximately satisfy (20)–(22). Therefore, after the convergence of the D-TOAMP with the tractable surrogate function in (26), we not only obtain an approximate stationary solution $\boldsymbol{\beta}_t^*$ of (19), but also the associated (approximate) marginal conditional posterior $p(x_{t,m}|\mathbf{y}^{(t)},\boldsymbol{\beta}_t^*)\approx\hat{p}(x_{t,m}|\mathbf{y}^{(t)},\boldsymbol{\beta}_t^*)$.

The detailed update expression for $\boldsymbol{\beta}_t$ could be found in Appendix A.

B. D-TOAMP-E Step

We first give an overview of the OAMP technique that will be used in the algorithm design in this subsection. Then, we elaborate the modules of the D-TOAMP-E step and the message passing in Module B at each time slot in the D-TOAMP-E step.

1) Overview of OAMP: OAMP proposed in [19]–[21] is a variation of the well-known approximate message passing (AMP) [28]. OAMP can handle a wide range of partial orthogonal sensing matrices, and it is shown to achieve a better performance than AMP. Consider the linear observation model

$$\mathbf{y}=\mathbf{F}\mathbf{x}+\mathbf{n}, \quad (27)$$

where $\mathbf{x}\in\mathbb{C}^{Q\times 1}$ is a sparse signal to be estimated, $\mathbf{y}\in\mathbb{C}^{L\times 1}$ is the received signal, and $\mathbf{n}\sim\mathcal{CN}(\mathbf{0},\sigma_e^2\boldsymbol{I})$ is the Gaussian noise, $\mathbf{F}\in\mathbb{C}^{L\times Q}$ is a partial orthogonal matrix. The entries of the sparse signal $\mathbf{x}$ are assumed to be i.i.d., with the j-th entry following the Bernoulli-Gaussian distribution:

$$x_j\sim\begin{cases}0 & \text{probability}=1-\lambda,\\ \mathcal{CN}(0,\xi) & \text{probability}=\lambda.\end{cases} \quad (28)$$

OAMP is designed to recover the i.i.d. sparse signal $\mathbf{x}$ from the linear observation model in (27).

OAMP contains two modules: Module A is a linear MMSE (LMMSE) estimator based on the observation and the messages from Module B; Module B performs MMSE estimator that combines the i.i.d sparse prior in (28) and the messages from Module A. The extrinsic output [29] of a module is fed to the other module as a prior input. The two modules are executed iteratively until convergence. At the end, the estimation of $\mathbf{x}$ is given based on the posterior output of Module B.

Specifically, for Module A, it is based on the assumption that the entries of $\mathbf{x}$ are i.i.d with a prior mean $\mathbf{x}_A^{pri}$ and variance v_A^{pri}, where $\mathbf{x}_A^{pri}$ and v_A^{pri} are the messages passed from Module B. Then under this assumption, the LMMSE estimate and the mean-square error (MSE) of $\mathbf{x}$ based on model (27) are respectively given by [19]

$$\mathbf{x}_A^{post}=\mathbf{x}_A^{pri}+\frac{v_A^{pri}}{v_A^{pri}+\sigma_e^2}\mathbf{F}^H(\mathbf{y}-\mathbf{F}\mathbf{x}_A^{pri}), \quad (29)$$

$$v_A^{post}=v_A^{pri}-\frac{L}{Q}\cdot\frac{(v_A^{pri})^2}{v_A^{pri}+\sigma_e^2}. \quad (30)$$

The extrinsic LMMSE estimate and the MSE of $\mathbf{x}$ can be computed by [19]

$$\mathbf{x}_B^{pri}=\mathbf{x}_A^{ext}=v_A^{ext}\left(\frac{\mathbf{x}_A^{post}}{v_A^{post}}-\frac{\mathbf{x}_A^{pri}}{v_A^{pri}}\right), \quad (31)$$

$$v_B^{pri}=v_A^{ext}=\left(\frac{1}{v_A^{post}}-\frac{1}{v_A^{pri}}\right)^{-1}, \quad (32)$$

which is passed to Module B as its input messages.

For Module B, it is based on the assumption that $\mathbf{x}_B^{pri}$ is modeled as an additive white Gaussian noise (AWGN) observation of $\mathbf{x}$, i.e.,

$$\mathbf{x}_B^{pri}=\mathbf{x}+\mathbf{z}, \quad (33)$$

where $\mathbf{z}\sim\mathcal{CN}(0,v_B^{pri}\boldsymbol{I})$ is independent of $\mathbf{x}$. Based on this assumption, the posterior mean and variance can be respectively calculated as [19]

$$x_{B,j}^{post}=\mathrm{E}\left(x_j|\mathbf{x}_B^{pri}\right)=\mathrm{E}\left(x_j|x_{B,j}^{pri}\right), \quad (34)$$

$$v_B^{post}=\frac{1}{Q}\sum_{j=1}^{Q}\mathrm{Var}\left(x_j|x_{B,j}^{pri}\right)=\frac{1}{Q}\sum_{j=1}^{Q}\mathrm{E}\left(\left|x_j-\mathrm{E}\left(x_j|x_{B,j}^{pri}\right)\right|^2\right), \quad (35)$$

where $x_{B,j}^{post}$ and $x_{B,j}^{pri}$ denote the j-th entry of $\mathbf{x}_B^{post}$ and $\mathbf{x}_B^{pri}$. $\mathrm{E}(\cdot)$ is with respect to the posterior distribution of $\mathbf{x}$, which is given by $p(x_j|x_{B,j}^{pri})\propto p(x_{B,j}^{pri}|x_j)p(x_j)$, where $p(x_j)$ is given in (28). Based on (33), $p(x_{B,j}^{pri}|x_j)$ is given by $\mathcal{CN}(x_{B,j}^{pri};x_j,v_B^{pri})$. Note that as the entries of $\mathbf{x}$ are prior independent (from (28)) and according to the assumption in (33), the entries of $\mathbf{x}$ are also posterior independent. The extrinsic mean

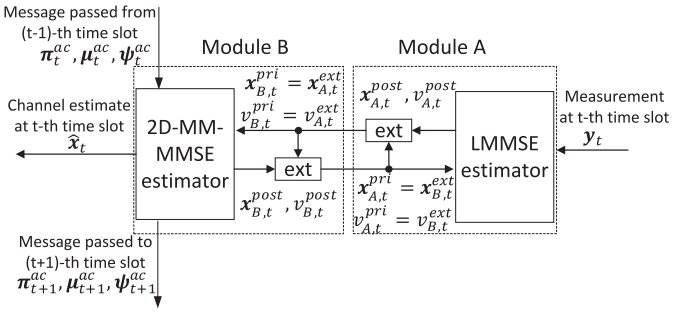


Fig. 7. Modules of the D-TOAMP algorithm.

and variance of $\mathbf{x}$ can be computed by [19]

$$\mathbf{x}_A^{pri} = \mathbf{x}_B^{ext} = v_B^{ext}\left(\frac{\mathbf{x}_B^{post}}{v_B^{post}} - \frac{\mathbf{x}_B^{pri}}{v_B^{pri}}\right), \tag{36}$$

$$v_A^{pri} = v_B^{ext} = \left(\frac{1}{v_B^{post}} - \frac{1}{v_B^{pri}}\right)^{-1}. \tag{37}$$

2) Modules of the D-TOAMP-E Step Within a Time Slot: At each time slot t, for given $\boldsymbol{\beta}_t$, the D-TOAMP-E step contains two modules (as shown in Fig. 7): Module A is a LMMSE estimator based on the current observation $\mathbf{y}_t$ and messages from Module B. Module B performs MMSE estimation that combines the 2D-MM channel prior in (9), the messages from Module A and the messages passed from the last time slot (for $t > 1$). The two modules are executed iteratively until convergence. Because in the D-TOAMP-E step, $\boldsymbol{\beta}_t$ is fixed, we will omit the argument $\boldsymbol{\beta}_t$ in the probability expressions for simplicity.

We now provide more details of each module. In Module A, the channel vector $\mathbf{x}_t$ is estimated based on the observation $\mathbf{y}_t$ with a prior distribution $\mathcal{CN}(\mathbf{x}_t; \mathbf{x}_{A,t}^{pri}, v_{A,t}^{pri}\boldsymbol{I})$, where $\mathbf{x}_{A,t}^{pri}$ and $v_{A,t}^{pri}$ are the extrinsic mean and variance, respectively, from the 2D-MM-MMSE estimator that will be elaborated in detail later. Then the posterior distribution of $\mathbf{x}_t$ is still complex Gaussian with mean and variance given by

$$\mathbf{x}_{A,t}^{post} = \mathbf{x}_{A,t}^{pri} + \frac{v_{A,t}^{pri}}{v_{A,t}^{pri} + \sigma_e^2}\mathbf{F}_t\left(\boldsymbol{\beta}_t\right)^H\left(\mathbf{y}_t - \mathbf{F}_t\left(\boldsymbol{\beta}_t\right)\mathbf{x}_{A,t}^{pri}\right) \tag{38}$$

and

$$v_{A,t}^{post} = v_{A,t}^{pri} - \frac{P}{M}\cdot\frac{(v_{A,t}^{pri})^2}{v_{A,t}^{pri} + \sigma_e^2}, \tag{39}$$

respectively. After that, we need to calculate the extrinsic message passing [29], which can decorrelate the input and output messages of the estimator. The extrinsic distribution of $\mathbf{x}_t$ satisfies

$$\begin{aligned}&\mathcal{CN}(\mathbf{x}_t; \mathbf{x}_{A,t}^{post}, v_{A,t}^{post}\boldsymbol{I})\\ &\propto \mathcal{CN}(\mathbf{x}_t; \mathbf{x}_{A,t}^{pri}, v_{A,t}^{pri}\boldsymbol{I})\mathcal{CN}(\mathbf{x}_t; \mathbf{x}_{A,t}^{ext}, v_{A,t}^{ext}\boldsymbol{I}).\end{aligned} \tag{40}$$

Therefore, the extrinsic mean and variance are respectively given by

$$\mathbf{x}_{B,t}^{pri} = \mathbf{x}_{A,t}^{ext} = v_{A,t}^{ext}\left(\frac{\mathbf{x}_{A,t}^{post}}{v_{A,t}^{post}} - \frac{\mathbf{x}_{A,t}^{pri}}{v_{A,t}^{pri}}\right), \tag{41}$$

$$v_{B,t}^{pri} = v_{A,t}^{ext} = \left(\frac{1}{v_{A,t}^{post}} - \frac{1}{v_{A,t}^{pri}}\right)^{-1}. \tag{42}$$

In Module B, the extrinsic calculation is similar to that in Module A, but the 2D-MM-MMSE estimator is more complicated.

Challenge 3: MMSE estimator design for the 2D-MM priors

In OAMP [19], the MMSE estimator was designed for i.i.d. prior. However, in this paper, the massive MIMO channels with 2D dynamic sparsity are not i.i.d distributed. The MMSE estimator needs to be redesigned based on the proposed 2D-MM channel prior. Therefore, the standard MMSE estimator for i.i.d. prior cannot be applied and we need to extend the MMSE estimator in order to exploit the 2D dynamic sparsity structure in the massive MIMO channels in (9). The details of the 2D-MM-MMSE estimator and the corresponding extrinsic update are presented in Section V-B3.

3) Message Passing in Module B (2D-MM-MMSE Estimator): In this subsection, we explain the details of Module B for 2D-MM channel priors at time slot t, $1 \le t \le T$. A basic assumption is to model $\mathbf{x}_{B,t}^{pri}$, the extrinsic mean from the LMMSE estimator as given in (41), as an AWGN observation [19], i.e.,

$$\mathbf{x}_{B,t}^{pri} = \mathbf{x}_t + \mathbf{z}_t, \tag{43}$$

where $\mathbf{z}_t \sim \mathcal{CN}(\mathbf{0}, v_{B,t}^{pri}\boldsymbol{I})$ is independent of $\mathbf{x}_t$, and $v_{B,t}^{pri}$ is the extrinsic variance from the LMMSE estimator as given in (42). Similar assumptions have been widely used in message-passing-based iterative signal recovery algorithms [13], [19], [20], [28]. A formal proof of Assumption (43) has been provided in [30] for the AMP with an i.i.d. Gaussian measurement matrix. Extensive simulations have also been conducted in [13], [19] and [20] to verify the validity of Assumption (43) for the OAMP. The main advantage of replacing the original observation model in (16) with the approximate AWGN observation model in (43) is that the per iteration complexity of the message passing algorithm can be reduced from $O\left(PM\right)$ to only $O(M)$.

Denote the collection of measurement vectors in AWGN observation model as $\mathbf{x}_B^{pri(T)} = \{\mathbf{x}_{B,t}^{pri}\}_{t=1}^T$. Under Assumption (43), the factor graph of joint distribution $p(\mathbf{x}_B^{pri(T)}, \mathbf{x}^{(T)}, \mathbf{s}^{(T)}, \boldsymbol{\theta}^{(T)})$, denoted by $\mathcal{G}$, is shown in Fig. 8, where the function expression of each factor node is listed in Table I. At time slot t, the factor graph denoted by $\mathcal{G}_t$ can be decomposed into three parts: the channel coefficient subgraph $\mathcal{G}_{c,t}$ (which represents the AWGN measurement model (43)); the hidden support subgraph $\mathcal{G}_{s,t}$ (which represents the 2D-MM $\mathbf{s}_t$); and the hidden value subgraph $\mathcal{G}_{\theta,t}$ (which represents the Gauss-Markov $\boldsymbol{\theta}_t$). Module B aims at

TABLE I
FACTORS, DISTRIBUTIONS AND FUNCTIONAL FORMS IN OUR SIGNAL MODEL

Factor	Distribution	Functional form
$g_{t,m}\left(x_{t,m}, x_{B,t,m}^{pri}\right)$	$p\left(x_{B,t,m}^{pri}\middle\vert x_{t,m}\right)$	$\mathcal{CN}\left(x_{t,m}; x_{B,t,m}^{pri}, v_{B,t}^{pri}\right)$
$f_{t,m}\left(x_{t,m}, s_{t,m}, \theta_{t,m}\right)$	$p\left(x_{t,m}\middle\vert s_{t,m}, \theta_{t,m}\right)$	$\delta\left(x_{t,m} - \theta_{t,m} s_{t,m}\right)$
$h_{1,1}\left(s_{1,1}\right)$	$p\left(s_{1,1}\right)$	$(1-\lambda)^{1-s_{1,1}}\left(\lambda\right)^{s_{1,1}}$
$h_{1,m}\left(s_{1,m}, s_{1,m-1}\right)$	$p\left(s_{1,m}\middle\vert s_{1,m-1}\right)$	$\begin{cases}\left(\rho_{01}^{S}\right)^{s_{1,m}}\left(1-\rho_{01}^{S}\right)^{1-s_{1,m}}, & s_{1,m-1}=0\\ \left(1-\rho_{10}^{S}\right)^{s_{1,m}}\left(\rho_{10}^{S}\right)^{1-s_{1,m}} & s_{1,m-1}=1\end{cases}$
$h_{t,1}\left(s_{t,1}, s_{t-1,1}\right)$	$p\left(s_{t,1}\middle\vert s_{t-1,1}\right)$	$\begin{cases}\left(\rho_{01}^{T}\right)^{s_{t,1}}\left(1-\rho_{01}^{T}\right)^{1-s_{t,1}}, & s_{t-1,1}=0\\ \left(1-\rho_{10}^{T}\right)^{s_{t,1}}\left(\rho_{10}^{T}\right)^{1-s_{t,1}} & s_{t-1,1}=1\end{cases}$
$h_{t,m}\left(s_{t,m}, s_{t,m-1}, s_{t-1,m}\right)$	$p\left(s_{t,m}\middle\vert s_{t,m-1}, s_{t-1,m}\right)$	$\left(\rho_{bc1}\right)^{s_{t,m}}\left(1-\rho_{bc1}\right)^{1-s_{t,m}}, s_{t,m-1}=b, s_{t-1,m}=c, b, c\in\{0,1\}$
$d_{1,m}\left(\theta_{1,m}\right)$	$p\left(\theta_{1,m}\right)$	$\mathcal{CN}\left(\theta_{1,m}; \zeta, \sigma^2\right)$
$d_{t,m}\left(\theta_{t,m}, \theta_{t-1,m}\right)$	$p\left(\theta_{t,m}\middle\vert \theta_{t-1,m}\right)$	$\mathcal{CN}\left(\theta_{t,m}; \left(1-\alpha\right)\theta_{t-1,m}+\alpha\zeta, \alpha^2\kappa\right)$

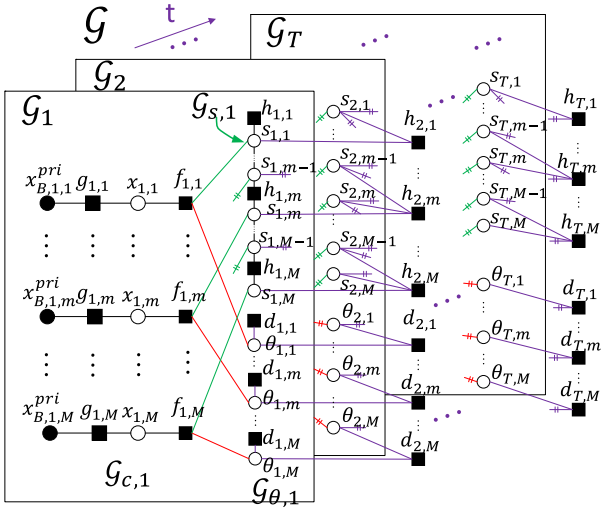


Fig. 8. Factor graph of the D-TOAMP.

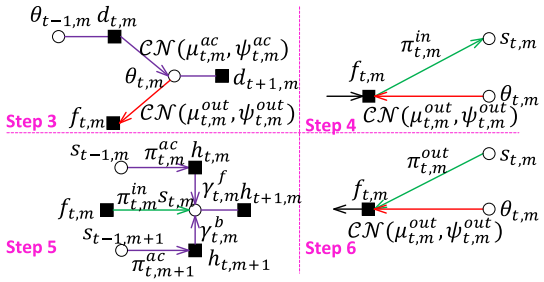


Fig. 9. Message passing of Step 3–6 in Algorithm 1.

calculating the posterior distributions $\{p(x_{t,m}|\mathbf{x}_B^{pri(t)})\}$ using the sum-product message passing rule.

The message passing order at time slot t is elaborated in Fig. 9. This figure also introduces the notations that we adopt to parameterize the different messages. For hidden support variables $s_{t,m}$ with a binary distribution, the associated message is represented by its nonzero probability, e.g., $\pi_{t,m}^{in} = \nu_{f_{t,m}\to s_{t,m}}\left(s_{t,m}=1\right)$. For hidden value variable $\theta_{t,m}$ with a complex Gaussian distribution, the associated message is parameterized using its mean and variance, i.e., $\nu_{\theta_{t,m}\to f_{t,m}}\left(\theta_{t,m}\right) = \mathcal{CN}(\theta_{t,m}; \mu_{t,m}^{out}, \psi_{t,m}^{out})$. Note that at the first time slot, the variable nodes $s_{t-1,m}, \theta_{t-1,m}$ and associated edges will be removed. At time slot T, the factor nodes $h_{t+1,m}, d_{t+1,m}$ and associated edges will be removed. The details are elaborated as follows.

Step 3 in Algorithm 1: Based on the prior information passed from the last time slot $\nu_{d_{t,m}\to\theta_{t,m}} = \mathcal{CN}(\theta_{t,m}; \mu_{t,m}^{ac}, \psi_{t,m}^{ac})$, the message passed from variable node $\theta_{t,m}$ to factor node $f_{t,m}$ can be calculated as

$$\nu_{\theta_{t,m}\to f_{t,m}}\left(\theta_{t,m}\right) = \mathcal{CN}\left(\theta_{t,m}; \mu_{t,m}^{out}, \psi_{t,m}^{out}\right), \tag{44}$$

where

$$\left(\mu_{t,m}^{out}, \psi_{t,m}^{out}\right) = \left(\mu_{t,m}^{ac}, \psi_{t,m}^{ac}\right). \tag{45}$$

Note that since the proposed algorithm is a recursive algorithm, there is no message passed from $d_{t+1,m}$ due to causality. If $t=1$, we set $(\mu_{1,m}^{ac}, \psi_{1,m}^{ac}) = (\zeta, \sigma^2)$.

Step 4 in Algorithm 1: According to the sum-product rule, the message from variable node $x_{t,m}$ to factor node $f_{t,m}$ is

$$\nu_{x_{t,m}\to f_{t,m}}\left(x_{t,m}\right) = \mathcal{CN}\left(x_{t,m}; x_{B,t,m}^{pri}, v_{B,t}^{pri}\right). \tag{46}$$

The message from factor node $f_{t,m}$ to variable node $s_{t,m}$ is

$$\begin{aligned}&\nu_{f_{t,m}\to s_{t,m}}\left(s_{t,m}\right)\\ &= \int_x\int_\theta \nu_{\theta_{t,m}\to f_{t,m}}\left(\theta\right)\nu_{x_{t,m}\to f_{t,m}}\left(x\right)\delta\left(x-\theta s_{t,m}\right)d\theta dx\\ &= \pi_{t,m}^{in}\,\delta\left(s_{t,m}-1\right) + \left(1-\pi_{t,m}^{in}\right)\delta\left(s_{t,m}\right),\end{aligned} \tag{47}$$

where

$$\pi_{t,m}^{in} = \left(1 + \frac{\mathcal{CN}\left(0; x_{B,t,m}^{pri}, v_{B,t}^{pri}\right)}{\mathcal{CN}\left(0; x_{B,t,m}^{pri} - \mu_{t,m}^{out}, v_{B,t}^{pri} + \psi_{t,m}^{out}\right)}\right)^{-1}. \tag{48}$$

Step 5 in Algorithm 1: A forward-backward message passing is performed over the 2D-MM of $\mathbf{s}_t$ given input messages $\{\nu_{f_{t,m}\to s_{t,m}}(s_{t,m})\}$, $\{\nu_{s_{t-1,m}\to h_{t,m}}(s_{t-1,m})\}$. Details are summarized in Algorithm 2. Note that there is no message passed from $h_{t+1,m}$ due to causality.

Step 6 in Algorithm 1: After this, the message from variable node $s_{t,m}$ to factor node $f_{t,m}$ can be calculated as

$$\nu_{s_{t,m}\to f_{t,m}}\left(s_{t,m}\right) = \pi_{t,m}^{out}\,\delta\left(s_{t,m}-1\right) + \left(1-\pi_{t,m}^{out}\right)\delta\left(s_{t,m}\right), \tag{49}$$

Algorithm 2: Channel Support Estimation Procedure.

Input: $\pi_{t,m}^{ac}, \pi_{t,m}^{in}, \forall m$
Output: $\pi_{t,m}^{out}, \forall m$
If $t = 1$
Initialize: $\gamma_{t,1}^{f} = \lambda$, $\gamma_{t,M}^{b} = \frac{1}{2}$
for $m = 2, \ldots, M$
$\gamma_{t,m}^{f} = \frac{\rho_{01}^{S}(1-\pi_{t,m-1}^{in})(1-\gamma_{t,m-1}^{f})+\rho_{11}^{S}\pi_{t,m-1}^{in}\gamma_{t,m-1}^{f}}{(1-\pi_{t,m-1}^{in})(1-\gamma_{t,m-1}^{f})+\pi_{t,m-1}^{in}\gamma_{t,m-1}^{f}}$.
for $m = 1, \ldots, M-1$
$\gamma_{t,m}^{b} = \frac{\rho_{10}^{S}(1-\pi_{t,m+1}^{in})(1-\gamma_{t,m+1}^{b})+(1-\rho_{10}^{S})\pi_{t,m+1}^{in}\gamma_{t,m+1}^{b}}{(\rho_{00}^{S}+\rho_{10}^{S})(1-\pi_{t,m+1}^{in})(1-\gamma_{t,m+1}^{b})+(\rho_{11}^{S}+\rho_{01}^{S})\pi_{t,m+1}^{in}\gamma_{t,m+1}^{b}}$.
Else If $t > 1$
Initialize: $\gamma_{t,1}^{f} = \rho_{01}^{T}(1-\pi_{t,1}^{ac}) + (1-\rho_{10}^{T})\pi_{t,1}^{ac}$, $\gamma_{t,M}^{b} = \frac{1}{2}$.
for $m = 2, \ldots, M$
$\gamma_{t,m}^{f} = \frac{\rho_{111}\pi_{t,m}^{ac}+\rho_{101}(1-\pi_{t,m}^{ac})}{1+((\pi_{t,m-1}^{in})^{-1}-1)((\gamma_{t,m-1}^{f})^{-1}-1)} + \frac{\rho_{011}\pi_{t,m}^{ac}+\rho_{001}(1-\pi_{t,m}^{ac})}{((\pi_{t,m-1}^{in})^{-1}-1)^{-1}((\gamma_{t,m-1}^{f})^{-1}-1)^{-1}+1}$.
for $m = 1, \ldots, M-1$
$\gamma_{t,m}^{b} = \frac{1}{1+\gamma}$, where $\gamma = \frac{\left(1-\pi_{t,m+1}^{in}-\gamma_{t,m+1}^{b}\right)\left[\rho_{010}\pi_{t,m+1}^{ac}+\rho_{000}\left(1-\pi_{t,m+1}^{ac}\right)\right]+\pi_{t,m+1}^{in}\gamma_{t,m+1}^{b}}{\left(1-\pi_{t,m+1}^{in}-\gamma_{t,m+1}^{b}\right)\left[\rho_{110}\pi_{t,m+1}^{ac}+\rho_{100}\left(1-\pi_{t,m+1}^{ac}\right)\right]+\pi_{t,m+1}^{in}\gamma_{t,m+1}^{b}}$.
end
Then $\pi_{t,m}^{out} = \frac{\gamma_{t,m}^{f}\gamma_{t,m}^{b}}{\gamma_{t,m}^{f}\gamma_{t,m}^{b}+\left(1-\gamma_{t,m}^{f}\right)\left(1-\gamma_{t,m}^{b}\right)}$.

where $\pi_{t,m}^{out}$ is given by the output of Algorithm 2. The message from factor node $f_{t,m}$ back to variable node $x_{t,m}$ is

$$\begin{aligned}&\nu_{f_{t,m}\to x_{t,m}}(x_{t,m})\\&= \pi_{t,m}^{out}\mathcal{CN}\left(x_{t,m};\mu_{t,m}^{out},\psi_{t,m}^{out}\right)+\left(1-\pi_{t,m}^{out}\right)\delta(x_{t,m}).\end{aligned} \tag{50}$$

After calculating the updated messages $\left\{\nu_{f_{t,m}\to x_{t,m}}\right\}$, the posterior distributions are given by

$$p\left(x_{t,m}|\mathbf{x}_B^{pri(t)}\right) \propto \nu_{f_{t,m}\to x_{t,m}}(x_{t,m})\,\nu_{g_{t,m}\to x_{t,m}}(x_{t,m}), \tag{51}$$

where $\nu_{g_{t,m}\to x_{t,m}}(x_{t,m}) = \mathcal{CN}(x_{t,m};x_{B,t,m}^{pri},v_{B,t}^{pri})$. Then the posterior mean and variance can be respectively calculated as

$$x_{B,t,m}^{post} = \mathrm{E}\left(x_{t,m}|\mathbf{x}_B^{pri(t)}\right) = \int_{x_{t,m}} x_{t,m}\,p\left(x_{t,m}|\mathbf{x}_B^{pri(t)}\right) \tag{52}$$

and

$$\begin{aligned}v_{B,t}^{post} &= \frac{1}{M}\sum_{m=1}^{M}\mathrm{Var}\left(x_{t,m}|\mathbf{x}_B^{pri(t)}\right)\\&= \frac{1}{M}\sum_{m=1}^{M}\int_{x_{t,m}}\left|x_{t,m}-\mathrm{E}\left(x_{t,m}|\mathbf{x}_B^{pri(t)}\right)\right|^2 p\left(x_{t,m}|\mathbf{x}_B^{pri(t)}\right).\end{aligned} \tag{53}$$

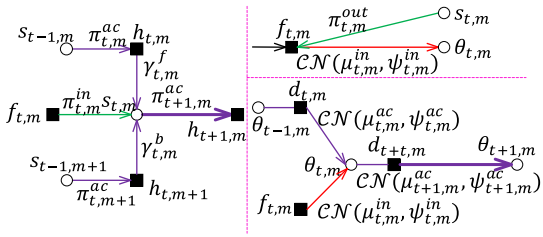


Fig. 10. Message passing across time slots.

Based on the derivation in [19], the corresponding extrinsic update can be calculated as

$$\mathbf{x}_{A,t}^{pri} = \mathbf{x}_{B,t}^{ext} = v_{A,t}^{pri}\left(\frac{\mathbf{x}_{B,t}^{post}}{v_{B,t}^{post}}-\frac{\mathbf{x}_{B,t}^{pri}}{v_{B,t}^{pri}}\right), \tag{54}$$

$$v_{A,t}^{pri} = v_{B,t}^{ext} = \left(\frac{1}{v_{B,t}^{post}}-\frac{1}{v_{B,t}^{pri}}\right)^{-1}. \tag{55}$$

C. Message Passing Across Time Slots

After the convergence of the message passing over $\mathcal{G}_t$, we forward the latest messages about $\mathbf{s}_t$ and $\boldsymbol{\theta}_t$ to the next time slot to provide prior information about $\mathbf{s}_{t+1}$ and $\boldsymbol{\theta}_{t+1}$. The message passing procedure is shown in Fig. 10. We firstly calculate channel support prior information passed to the next time slot $\nu_{s_{t,m}\to h_{t+1,m}}(s_{t,m})$ as follows:

$$\begin{aligned}&\nu_{s_{t,m}\to h_{t+1,m}}(s_{t,m})\\&= \nu_{h_{t,m}\to s_{t,m}}(s_{t,m})\nu_{h_{t,m+1}\to s_{t,m}}(s_{t,m})\nu_{f_{t,m}\to s_{t,m}}(s_{t,m})\\&= \pi_{t+1,m}^{ac}\delta(s_{t,m}-1)+(1-\pi_{t+1,m}^{ac})\delta(s_{t,m}),\end{aligned} \tag{56}$$

where $\pi_{t+1,m}^{ac}$ is given by

$$\pi_{t+1,m}^{ac} = \frac{\gamma_{t,m}^{f}\gamma_{t,m}^{b}\pi_{t,m}^{in}}{\gamma_{t,m}^{f}\gamma_{t,m}^{b}\pi_{t,m}^{in}+\left(1-\gamma_{t,m}^{f}\right)\left(1-\gamma_{t,m}^{b}\right)\left(1-\pi_{t,m}^{in}\right)}.$$

In order to calculate the hidden value vector prior information passed to next time slot, we firstly calculate the message from factor node $f_{t,m}$ to variable node $\theta_{t,m}$ based on the latest information as follows:

$$\begin{aligned}&\nu_{f_{t,m}\to\theta_{t,m}}^{\text{exact}}(\theta_{t,m})\\&= \int_x\sum_s \nu_{s_{t,m}\to f_{t,m}}(s)\,\nu_{x_{t,m}\to f_{t,m}}(x)\,\delta(x-\theta_{t,m}s)\,dx\\&= \pi_{t,m}^{out}\mathcal{CN}\left(\theta_{t,m};x_{B,t,m}^{pri},v_{B,t}^{pri}\right)\\&\quad+\left(1-\pi_{t,m}^{out}\right)\mathcal{CN}\left(0;x_{B,t,m}^{pri},v_{B,t}^{pri}\right).\end{aligned} \tag{57}$$

This is an inappropriate message, because $\mathcal{CN}(0;x_{B,t,m}^{pri},v_{B,t}^{pri})$ which is irrelevant to $\theta_{t,m}$ prevents us from normalizing it. Intuitively, $\nu_{f_{t,m}\to\theta_{t,m}}(\theta_{t,m})$ conveys the information about hidden values $\theta_{t,m}$ based on the channel support $s_{t,m}$ and actual channel coefficient $x_{t,m}$. If $s_{t,m}=0$, then by (8), $x_{t,m}=0$. We can not obtain any useful information about $\theta_{t,m}$, which makes

$\theta_{t,m}$ unobservable. The constant term in (57) reveals the uncertainty caused by this unobservability through an infinitely broad and uninformative distribution of $\theta_{t,m}$. A similar problem also occurs in [18], we adopt the similar approach to solve it by introducing a threshold. The resulting proper message is given by

$$\nu_{f_{t,m}\to\theta_{t,m}}\left(\theta_{t,m}\right) = \mathcal{CN}\left(\theta_{t,m}; \mu^{in}_{t,m}, \psi^{in}_{t,m}\right) \tag{58}$$

where

$$\left(\mu^{in}_{t,m}, \psi^{in}_{t,m}\right) = \begin{cases} \left(\frac{1}{\epsilon}x^{pri}_{B,t,m}, \frac{1}{\epsilon^2}v^{pri}_{B,t}\right) & \pi^{out}_{t,m} \le \mathrm{Thr}, \\ \left(x^{pri}_{B,t,m}, v^{pri}_{B,t}\right) & \pi^{out}_{t,m} > \mathrm{Thr}, \end{cases} \tag{59}$$

the threshold Thr is slightly less than 1 and ϵ is close to 0. Then the prior information about the hidden value vector passed to the next time slot can be calculated as follows:

$$\nu_{d_{t+1,m}\to\theta_{t+1,m}}\left(\theta_{t+1,m}\right) = \mathcal{CN}\left(\theta_{t+1,m}; \mu^{ac}_{t+1,m}, \psi^{ac}_{t+1,m}\right), \tag{60}$$

where

$$\mu^{ac}_{t+1,m} = (1-\alpha)\left(\frac{\psi^{ac}_{t,m}\psi^{in}_{t,m}}{\psi^{ac}_{t,m}+\psi^{in}_{t,m}}\right)\left(\frac{\mu^{ac}_{t,m}}{\psi^{ac}_{t,m,n}}+\frac{\mu^{in}_{t,m}}{\psi^{in}_{t,m}}\right)+\alpha\zeta,$$

$$\psi^{ac}_{t+1,m} = (1-\alpha)^2\left(\frac{\psi^{ac}_{t,m}\psi^{in}_{t,m}}{\psi^{ac}_{t,m}+\psi^{in}_{t,m}}\right)+\alpha^2\kappa.$$

Finally, the overall D-TOAMP algorithm is summarized in Algorithm 1. Note that the proposed D-TOAMP can also be applied to a general array geometry at the BS, by replacing the array response matrix $\mathbf{A}(\boldsymbol{\beta})$ with $\mathbf{A}(\boldsymbol{\beta}, \hat{\boldsymbol{\varphi}})$, and adding an additional gradient update for $\hat{\boldsymbol{\varphi}}$. The details are omitted for conciseness.

D. Complexity Analysis

The computational complexity of the proposed algorithm is analyzed as follows.

- Complexity of Module A (LMMSE estimator): The computational complexity of LMMSE is dominated by the matrix multiplication, whose complexity is $\mathcal{O}(PM)$.
- Complexity of Module B (2D-MM-MMSE estimator): This module is used to handle the 2D-MM prior (capture the 2D dynamic sparsity of a MIMO channel). Since this module is just a simple sum-product algorithm over a tree graph, and each message could be parameterized by one or two variables, the complexity is very low, which is $\mathcal{O}(M)$.
- Complexity of parameter update: The complexity of updating the off-grid parameter $\boldsymbol{\beta}_t$ is $\mathcal{O}(PM^2)$ per iteration if the fixed stepsize is used.

This suggests the total computational requirement of the proposed method with off-grid model is $\mathcal{O}\left(PM^2\right)$ per iteration. Empirical evidence shows that the proposed method usually converges within 30 iterations. Fig. 11 shows the simulation time of various schemes versus the number of antennas for not exactly sparse signals (the AoDs do not exactly lie on the grid points). The details of each baseline scheme are introduced in Section VI. It shows that the complexity of the proposed D-TOAMP without grid refinement is comparable to the message-passing-based algorithms with i.i.d. or structured priors, such as OAMP [19], DCS-AMP [18] and Structured Turbo-CS [13], and is lower than that of some popular CS algorithms, such as Modified SP [5], SBL [22] which require matrix multiplication and matrix inversion with high computational complexity, and Burst-LASSO [12] which needs to solve a large dimensional optimization problem. With off-grid basis, the off-grid parameter updating will introduce more complexity in order to eliminate the grid mismatch. However, the off-grid based D-TOAMP could achieve much better performance compared to the D-TOAMP without considering the grid mismatch, as shown in Section VI.

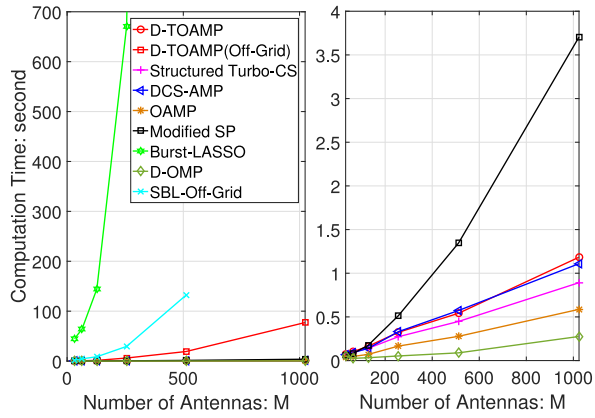


Fig. 11. Computation time of various schemes versus the number of antennas for not exactly sparse signals. Set $P/M = 0.18$, $T = 50$, SNR $= 15$ dB, $\lambda = 0.125$, $\rho^S_{01} = 0.025$, $\rho^S_{10} = 0.175$, $\rho^T_{01} = 0.025$, $\rho^T_{10} = 0.025$, $\rho_{111} = 0.9946$, $\rho_{001} = 0.0007$, $\rho_{011} = 0.5$, $\rho_{101} = 0.1078$, $\kappa = 1$, $\sigma = 0.23$, $\alpha = 0.1$, $\zeta = 0$. For Burst-LASSO and SBL baseline, we only simulate the case when $M = 64, 128, 256, 512$ due to their long computation time.

VI. Simulation Results

In this section, we evaluate the performance of the proposed algorithm under two widely used channel models which are not exactly sparse. The spatial channel model (SCM) [31] is developed in 3GPP/3GPP2 for low frequency bands (less than 6 GHz) and has been widely used to evaluate the performance of LTE systems. We consider urban microcell environment, and each channel realization consists of $N_c = 3$ random scattering clusters ranging from -40° to 40°, and each cluster contains $N_b = 10$ sub-paths concentrated in a 15° angular spread. The detailed realization could be found in [31]. Another realistic channel model is mm-SSCM proposed in [24] for high frequency bands (28–73 GHz). The mm-SSCM was developed based on 28- and 73-GHz ultrawideband propagation measurements in New York City, and has been shown to faithfully reproduce realistic impulse responses of measured urban channels. We consider 28 GHz outdoor environment. The number of AoD spatial lobes, the central AoD angle of each AoD spatial lobe and the angular spread are generated according to the distribution in [24]. We consider the following baseline algorithms:

D-OMP [3]: This algorithm proposes the differential orthogonal matching pursuit (D-OMP) algorithm to track a dynamic sparse channel by exploiting its temporal correlation.

Modified-SP [5]: This algorithm exploits the prior support and quality information provided by the previous estimated channel to enhance the current CE performance. We denote $\hat{\Omega}_{t-1}$ and $\hat{\Omega}_{t-2}$ as the estimated channel supports at the $(t-1)$-th and $(t-2)$-th time slots. Then at t-th time slot, $\mathcal{T}_0$, $\bar{s}$ and s_c in [5] are set to be $\hat{\Omega}_{t-1}$, $|\hat{\Omega}_{t-1}|$ and $|\hat{\Omega}_{t-1} \cap \hat{\Omega}_{t-2}|$, respectively.

OAMP [19]**:** The OAMP assumes i.i.d. sparse channel prior.

Burst-LASSO [12]: The Burst-LASSO exploits the bursty structure of the channel in the spatial domain.

Structured Turbo-CS [13]: The structured Turbo-CS exploits the clustered structure of channel in spatial domain.

DCS-AMP [18]: The DCS-AMP exploits the temporal correlation of sparse signal sequences.

SBL-Off-Grid [22]: The SBL-Off-Grid obtained the true AoD values of the massive MIMO channels by the sparse Bayesian learning (SBL) based algorithm.

To apply the proposed algorithm in a realistic channel, we update the statistic parameters $\boldsymbol{\rho} \triangleq \{\rho_{ba}^S, \rho_{ca}^T, \rho_{bca}, \alpha, \zeta, \kappa, \sigma^2\}$ where $a, b, c \in \{0, 1\}$ using the EM framework [18]. Specifically, the statistic parameters $\boldsymbol{\rho}$ are initialized using available prior knowledge. In each iteration of each time slot, they are updated based on the latest estimated marginal posterior distributions of $\mathbf{s}_t$, $\boldsymbol{\theta}_t$ and $\mathbf{x}_t$. The detailed EM update equations for $\boldsymbol{\rho}$ are omitted in this paper for conciseness, interested readers can refer to [18] for detailed derivations. Similarly, to apply the message-passing-based baselines in practical channel tracking, the channel statistical parameters ($\{\lambda, \zeta, \sigma^2\}$ for OAMP, $\{\rho_{01}^S, \rho_{10}^S, \zeta, \sigma^2\}$ for Structured Turbo-CS, $\{\rho_{01}^T, \rho_{10}^T, \alpha, \zeta, \kappa, \sigma^2\}$ for DCS-AMP) are also updated by the EM framework. Both low-frequency and high-frequency MIMO systems will be considered. In the low-frequency massive MIMO system, the BS has $M = 128$ antennas and the SCM will be used to generate channels. In the high-frequency massive MIMO systems, the BS has $M = 256$ antennas and the mm-SSCM will be used to generate channels. We focus on the simulations for the ULA. For the baselines, we use the fixed DFT basis except for the SBL-Off-Grid baseline, which uses the off-grid basis, and the off-grid parameters are updated using the SBL. For the proposed D-TOAMP algorithm, we verify its performance for both with and without grid refinement. In the following simulation results, D-TOAMP (DFT) means the proposed D-TOAMP algorithm with fixed DFT basis, i.e., $\boldsymbol{\beta}_t$ in (4) is set to be $\mathbf{0}$; D-TOAMP (Off-Grid) means the proposed D-TOAMP algorithm with off-grid basis, i.e., $\boldsymbol{\beta}_t$ in (4) is updated based on (25). We set $\mathrm{Thr} = 1 - 10^{-2}, \epsilon = 10^{-7}$. The primary performance metric that we used in all of our experiments, which we refer to as the time-averaged normalized MSE (TNMSE), is defined as

$$\mathrm{TNMSE} \triangleq \frac{1}{T} \sum_{t=1}^{T} \frac{\|\hat{\mathbf{x}}_t - \mathbf{x}_t\|^2}{\|\mathbf{x}_t\|^2}, \tag{61}$$

where $\hat{\mathbf{x}}_t$ is the estimate of $\mathbf{x}_t$ at t-th time slot.

A. Impact of SNR

In Fig. 12 and Fig. 13, we compare the TNMSE performance of different algorithms versus SNR under the SCM and the mm-SSCM, respectively. For each channel model, we also consider the effect of the user velocity on the channel tracking performance. It can be seen that under each user velocity value, the proposed D-TOAMP achieves sufficient performance gain over all the baseline algorithms, under both less sparse SCM and more sparse mm-SSCM. Moreover, the off-grid based D-TOAMP could further improve the channel tracking performance by mitigating the off-grid leakage. This demonstrates that the proposed algorithm can effectively track the realistic dynamic channels in a massive MIMO system by exploiting the 2D dynamic sparsity of channels.

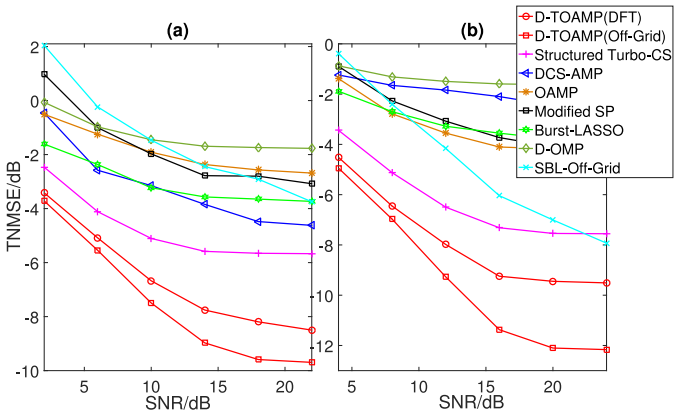


Fig. 12. TNMSE versus SNR under the SCM. Set $M = 128$, $P = 26$, and $T = 50$. (a) user velocity is 0.1 m/s; (b) user velocity is 1 m/s.

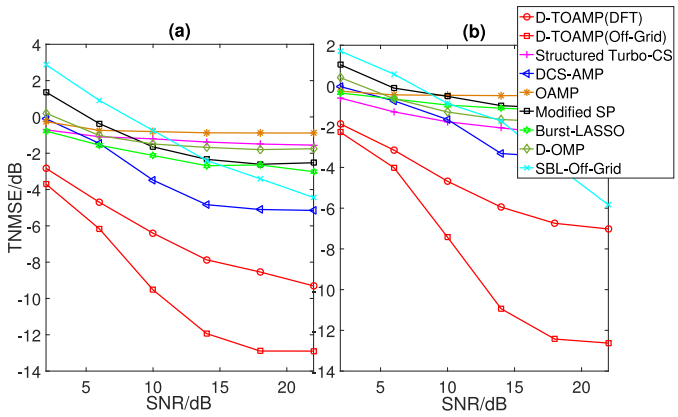


Fig. 13. TNMSE versus SNR under the mm-SSCM. Set $M = 256$, $P = 22$, and $T = 50$. (a) user velocity is 0.1 m/s; (b) user velocity is 1 m/s.

B. Impact of Pilot Number

In Fig. 14 and Fig. 15, we compare the TNMSE performance of different algorithms versus the number of pilot sequences P under the SCM and the mm-SSCM, respectively. For each channel model, we also consider the effect of the user velocity on the channel tracking performance. It can be observed that the TNMSE performance decreases as the number of pilots increases for all schemes. The proposed D-TOAMP algorithm can achieve large performance gain over various baselines for different user velocity values and different channel models. Moreover, the off-grid based D-TOAMP could further improve the channel

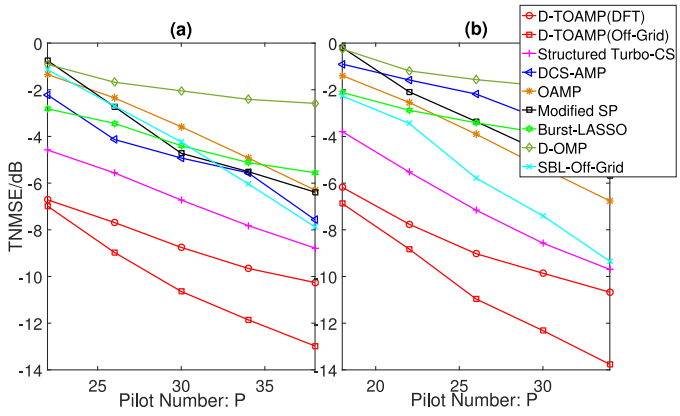


Fig. 14. TNMSE versus pilot number under the SCM. Set $M = 128$, SNR $= 15$ dB, and $T = 50$. (a) user velocity is 0.1 m/s; (b) user velocity is 1 m/s.

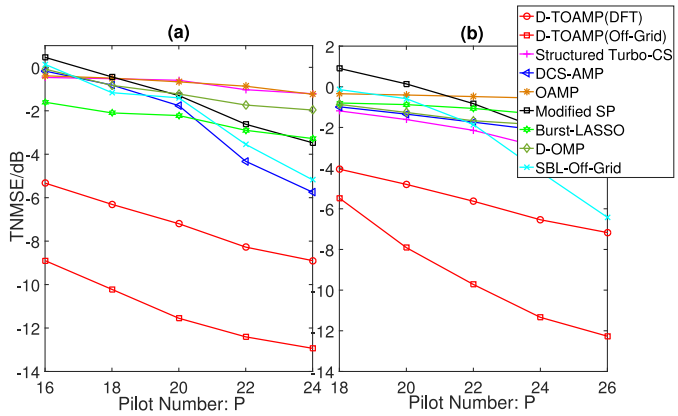


Fig. 15. TNMSE versus pilot number under the mmWave. Set $M = 256$, SNR $= 15$ dB, and $T = 50$. (a) user velocity is 0.1 m/s; (b) user velocity is 1 m/s.

tracking performance. This verifies that the proposed algorithm can accurately recover a time series of realistic channels with low pilot overhead.

VII. Conclusion

We consider the downlink channel tracking problem for a massive MIMO system. Firstly, we propose a statistical channel model called the 2D-MM to model the 2D dynamic sparsity of massive MIMO channels. Then we propose a D-TOAMP algorithm that can be used to recursively track sparse massive MIMO channels with 2D-MM prior. At each time slot, the message passing will be performed based on the prior information passed from the previous time slot and current measurements. Then we verify the superior performance of the proposed channel tracking algorithm under two realistic channel models: SCM and mm-SSCM. Extensive simulations show that the proposed off-grid based D-TOAMP algorithm derived from the 2D-MM channel prior can effectively exploit the 2D dynamic sparsity of practical massive MIMO channels to achieve significant gain over various baseline algorithms.

For clarify, we focus on frequency-flat fading channels in this paper. In frequency-selective fading channels, there is also structured sparsity in the delay/frequency domain [32]. An interesting future work is to propose a proper probability model to jointly capture the structured sparsity in the spatial/delay/frequency domain and the temporal domain. The proposed D-TOAMP framework can be extended to handle more practical frequency-selective fading channel tracking problem, by modifying the MMSE estimator (Module B) according to the new probability model. For multi-user massive MIMO channel tracking problem, another possibility is to further exploit the common support structure of the multi-user channels in spatial domain to enhance the channel tracking performance in multi-user massive MIMO system.

Appendix

A. Gradient Update for Off-Grid Parameters

After the D-TOAMP-E step, the posterior estimation of $\mathbf{x}_t$ at the i-th iteration is given by $\hat{p}\left(\mathbf{x}_t|\mathbf{y}^{(t)}, \boldsymbol{\beta}_t^i\right) = \mathcal{CN}(\mathbf{x}_{B,t}^{post}, v_{B,t}^{post}\boldsymbol{I})$ ($\mathbf{x}_{B,t}^{post}$ and $v_{B,t}^{post}$ are given by (52) and (53), respectively). Then the surrogate function $\hat{u}(\boldsymbol{\beta}_t; \boldsymbol{\beta}_t^i)$ in (26) can be calculated as

$$\begin{aligned}
&\hat{u}(\boldsymbol{\beta}_t; \boldsymbol{\beta}_t^i) \\
&\propto \mathrm{E}_{\hat{p}\left(\mathbf{x}_t|\mathbf{y}^{(t)}, \boldsymbol{\beta}_t^i\right)}\left[-\frac{1}{\sigma_e^2}\left\|\mathbf{y}_t - \mathbf{F}_t\left(\boldsymbol{\beta}_t\right)\mathbf{x}_t\right\|^2\right] \\
&\propto -\frac{1}{\sigma_e^2}\left(\left\|\mathbf{y}_t - \mathbf{F}_t\left(\boldsymbol{\beta}_t\right)\mathbf{x}_{B,t}^{post}\right\|^2\right. \\
&\qquad \left. + v_{B,t}^{post}\mathrm{tr}\left(\mathbf{F}_t\left(\boldsymbol{\beta}_t\right)\mathbf{F}_t\left(\boldsymbol{\beta}_t\right)^H\right)\right).
\end{aligned}$$

The derivative of $\hat{u}(\boldsymbol{\beta}_t; \boldsymbol{\beta}_t^i)$ w.r.t. $\boldsymbol{\beta}_t$ can be calculated as $\boldsymbol{\xi}_{\boldsymbol{\beta}_t}^{(i)} = \left[\xi^{(i)}\left(\beta_{t,1}\right), \ldots, \xi^{(i)}\left(\beta_{t,M}\right)\right]^T$, with

$$\begin{aligned}
&\xi^{(i)}\left(\beta_{t,m}\right) \\
&= 2\mathrm{Re}\left(\mathbf{a}'\left(\hat{\vartheta}_m + \beta_{t,m}\right)^H \mathbf{U}_t\mathbf{U}_t^H\mathbf{a}\left(\hat{\vartheta}_m + \beta_{t,m}\right)\right)c_1^{(i)} \\
&\quad + 2\mathrm{Re}\left(\mathbf{a}'\left(\hat{\vartheta}_m + \beta_{t,m}\right)^H \mathbf{U}_t\mathbf{c}_2^{(i)}\right), \qquad (62)
\end{aligned}$$

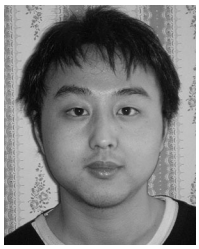
where $c_1^{(i)} = -\frac{1}{\sigma_e^2}(|x_{B,t,m}^{post}|^2 + v_{B,t}^{post})$, $\mathbf{c}_2^{(i)} = \frac{1}{\sigma_e^2}(x_{B,t,m}^{post})^\star\mathbf{y}_{t,-m}$, $\mathbf{y}_{t,-m} = \mathbf{y}_t - \mathbf{U}_t^H\sum_{j\neq m}\mathbf{a}(\hat{\vartheta}_j + \beta_{t,j})x_{B,t,j}^{post}$, $\mathbf{a}'(\hat{\vartheta}_m + \beta_{t,m}) = d\mathbf{a}(\hat{\vartheta}_m + \beta_{t,m})/d\beta_{t,m}$. Then the off-grid parameter $\boldsymbol{\beta}_t$ is updated as $\boldsymbol{\beta}_t^{i+1} = \boldsymbol{\beta}_t^i + \Delta^i \cdot \boldsymbol{\xi}_{\boldsymbol{\beta}_t^i}^{(i)}$.

References

[1] E. Larsson, O. Edfors, F. Tufvesson, and T. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.
[2] E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Trans. Telecommun.*, vol. 10, no. 6, pp. 585–595, 1999.
[3] X. Zhu, L. Dai, W. Dai, Z. Wang, and M. Moonen, "Tracking a dynamic sparse channel via differential orthogonal matching pursuit," in *Proc. IEEE Mil. Commun. Conf.*, Oct. 2015, pp. 792–797.
[4] X. Zhu, L. Dai, G. Gui, W. Dai, Z. Wang, and F. Adachi, "Structured matching pursuit for reconstruction of dynamic sparse channels," in *Proc. IEEE Global Commun. Conf.*, Dec. 2015, pp. 1–5.
[5] X. Rao and V. K. Lau, "Compressive sensing with prior support quality information and application to massive MIMO channel estimation with temporal correlation," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4914–4924, Sep. 2015.

[6] L. Dai and X. Gao, "Priori-aided channel tracking for millimeter-wave beamspace massive MIMO systems," in *Proc. URSI Asia-Pacific Radio Sci. Conf.*, Aug. 2016, pp. 1493–1496.
[7] Z. Gao, L. Dai, W. Dai, B. Shim, and Z. Wang, "Structured compressive sensing-based spatio-temporal joint channel estimation for FDD massive MIMO," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 601–617, Feb. 2016.
[8] 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.
[9] H. Yin, D. Gesbert, M. Filippou, and Y. Liu, "A coordinated approach to channel estimation in large-scale multiple-antenna systems," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 264–273, Feb. 2013.
[10] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
[11] C. R. Berger, Z. Wang, J. Huang, and S. Zhou, "Application of compressive sensing to sparse channel estimation," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 164–174, Nov. 2010.
[12] A. Liu, V. K. Lau, and W. Dai, "Exploiting burst-sparsity in massive MIMO with partial channel support information," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7820–7830, Nov. 2016.
[13] L. Chen, A. Liu, and X. Yuan, "Structured turbo compressed sensing for massive MIMO channel estimation using a Markov prior," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4635–4639, May 2018.
[14] O. Mehanna and N. D. Sidiropoulos, "Channel tracking and transmit beamforming with frugal feedback," *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6402–6413, Dec. 2014.
[15] X. Wang, J. Wang, L. He, C. Pan, and J. Song, "Basis expansion model based spectral efficient channel recovery scheme for spatial-temporal correlated massive MIMO systems," *IET Commun.*, vol. 11, no. 17, pp. 2621–2629, 2017.
[16] X. Liu, Y. Shi, J. Zhang, and K. B. Letaief, "Massive CSI acquisition in dense cloud-RAN with spatial and temporal prior information," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–6.
[17] J. Zhao, F. Gao, W. Jia, J. Zhao, and W. Zhang, "Channel tracking for massive MIMO systems with spatial-temporal basis expansion model," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–5.
[18] J. Ziniel and P. Schniter, "Dynamic compressive sensing of time-varying signals via approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5270–5284, Nov. 2013.
[19] J. Ma and L. Ping, "Orthogonal AMP," *IEEE Access*, vol. 5, pp. 2020–2033, 2017.
[20] J. Ma, X. Yuan, and L. Ping, "On the performance of turbo signal recovery with partial DFT sensing matrices," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1580–1584, Oct. 2015.
[21] J. Ma, X. Yuan, and L. Ping, "Turbo compressed sensing with partial DFT sensing matrix," *IEEE Signal Process. Lett.*, vol. 22, no. 2, pp. 158–161, Feb. 2015.
[22] J. Dai, A. Liu, and V. K. Lau, "FDD massive MIMO channel estimation with arbitrary 2D-array geometry," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2584–2599, May 2018.
[23] E. Fornasini, "2D Markov chains," *Linear Algebra Appl.*, vol. 140, pp. 101–127, 1990.
[24] M. K. Samimi and T. S. Rappaport, "3-D millimeter-wave statistical channel model for 5G wireless system design," *IEEE Trans. Microw. Theory Techn.*, vol. 64, no. 7, pp. 2207–2225, Jul. 2016.
[25] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc. Ser. B (Methodological)*, vol. 39, pp. 1–38, 1977.
[26] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. Berlin, Germany: Springer, 2006.
[27] C. J. Wu, "On the convergence properties of the EM algorithm," *Ann. Statist.*, vol. 11, pp. 95–103, 1983.
[28] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, no. 45, pp. 18914–18919, 2009.
[29] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.
[30] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 764–785, Feb. 2011.
[31] J. Salo *et al.*, "MATLAB implementation of the 3GPP spatial channel model (3GPP TR 25.996)," Jan. 2005. [Online]. Available: http://www.tkk.fi/Units/Radio/scm/
[32] Z. Gao, L. Dai, S. Han, I. Chih-Lin, Z. Wang, and L. Hanzo, "Compressive sensing techniques for next-generation wireless communications," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 144–153, Jun. 2018.

Lixiang Lian (S'16) received the B.Eng. degree in information and communication engineering from Zhejiang University, Hangzhou, China, in 2014. She is currently working toward the Ph.D. degree with the Department of Electronics and Communication Engineering, The Hong Kong University of Science and Technology, Hong Kong. Her research interests include wireless communication and compressive sensing.



An Liu (S'07–M'09–SM'17) received the B.S. and Ph.D. degrees in electrical engineering from Peking University, Beijing, China, in 2011 and 2004, respectively. From 2008 to 2010, he was a Visiting Scholar with the Department of Electrical, Computer, and Energy Engineering, University of Colorado at Boulder. He was a Postdoctoral Research Fellow from 2011 to 2013, a Visiting Assistant Professor in 2014, and a Research Assistant Professor from 2015 to 2017, with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology. He is currently a Distinguished Research Fellow with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China. His research interests include wireless communications, stochastic optimization, and compressive sensing.

Vincent K. N. Lau (SM'04–F'12) received the B.Eng. (Distinction first Hons.) degree from the University of Hong Kong, Hong Kong, in 1992 and the Ph.D. degree from Cambridge University, Cambridge, U.K., in 1997. He was with Bell Labs from 1997 to 2004 and the Department of Electronics and Communication Engineering, The Hong Kong University of Science and Technology (HKUST) in 2004. He is currently a Chair Professor and the Founding Director of Huawei-HKUST Joint Innovation Lab, HKUST. His current research interests includes robust and delay-optimal cross-layer optimization for MIMO/OFDM wireless systems, interference mitigation techniques for wireless networks, massive MIMO, M2M, and network control systems.