Channel estimation via gradient pursuit for mmWave massive MIMO systems with one-bit ADCs

In-soo Kim and Junil Choi* [ID]

Abstract

In millimeter wave (mmWave) massive multiple-input multiple-output (MIMO) systems, 1 bit analog-to-digital converters (ADCs) are employed to reduce the impractically high power consumption, which is incurred by the wide bandwidth and large arrays. In practice, the mmWave band consists of a small number of paths, thereby rendering sparse virtual channels. Then, the resulting maximum a posteriori (MAP) channel estimation problem is a sparsity-constrained optimization problem, which is NP-hard to solve. In this paper, iterative approximate MAP channel estimators for mmWave massive MIMO systems with 1 bit ADCs are proposed, which are based on the gradient support pursuit (GraSP) and gradient hard thresholding pursuit (GraHTP) algorithms. The GraSP and GraHTP algorithms iteratively pursue the gradient of the objective function to approximately optimize convex objective functions with sparsity constraints, which are the generalizations of the compressive sampling matching pursuit (CoSaMP) and hard thresholding pursuit (HTP) algorithms, respectively, in compressive sensing (CS). However, the performance of the GraSP and GraHTP algorithms is not guaranteed when the objective function is ill-conditioned, which may be incurred by the highly coherent sensing matrix. In this paper, the band maximum selecting (BMS) hard thresholding technique is proposed to modify the GraSP and GraHTP algorithms, namely, the BMSGraSP and BMSGraHTP algorithms, respectively. The BMSGraSP and BMSGraHTP algorithms pursue the gradient of the objective function based on the band maximum criterion instead of the naive hard thresholding. In addition, a fast Fourier transform-based (FFT-based) fast implementation is developed to reduce the complexity. The BMSGraSP and BMSGraHTP algorithms are shown to be both accurate and efficient, whose performance is verified through extensive simulations.

Keywords: MmWave, Massive MIMO, 1 bit ADC, MAP channel estimation, GraSP, GraHTP, CoSaMP, HTP, CS, BMS hard thresholding technique, FFT

1 Introduction

In millimeter wave (mmWave) massive multiple-input multiple-output (MIMO) systems, the wide bandwidth of the mmWave band in the range of 30–300 GHz offers a high data rate, which guarantees a significant performance gain [1–4]. However, the power consumption of analog-to-digital converters (ADCs) is scaled quadratically with the sampling rate and exponentially with the ADC resolution, which renders high-resolution ADCs impractical for mmWave systems [5]. To reduce the power consumption, low-resolution ADCs were suggested as a possible solution, which recently gained popularity [6–9]. Coarsely

quantizing the received signal using low-resolution ADCs results in an irreversible loss of information, which might cause a significant performance degradation. In this paper, we consider the extreme scenario of using 1 bit ADCs for mmWave systems.

In practice, the mmWave band consists of a small number of propagation paths, which results in sparse virtual channels. In the channel estimation point of view, sparse channels are favorable because the required complexity and measurements can be reduced. Sparsity-constrained channel distributions, however, cannot be described in closed forms, which makes it difficult to exploit Bayesian channel estimation. In [10, 11], channel estimators for massive MIMO systems with 1 bit ADCs were proposed, which account for the effect of the coarse quantization.

*Correspondence: junil@kaist.ac.kr
School of Electrical Engineering, KAIST, Daejeon, South Korea

The near maximum likelihood (nML) channel estimator [10] selects the maximizer of the likelihood function subject to the L^2-norm constraint as the estimate of the channel, which is solved using the projected gradient descent method [12]. However, the channel sparsity was not considered in [10]. In [11], the Bussgang linear minimum mean squared error (BLMMSE) channel estimator was derived by linearizing 1 bit ADCs based on the Bussgang decomposition [13]. The BLMMSE channel estimator is an LMMSE channel estimator for massive MIMO systems with 1 bit ADCs, whose assumption is that the channel is Gaussian. Therefore, the sparsity of the channel is not taken into account in [11] either.

To take the channel sparsity into account, iterative approximate MMSE estimators for mmWave massive MIMO systems with 1 bit ADCs were proposed in [14, 15]. The generalized expectation consistent signal recovery (GEC-SR) algorithm in [14] is an iterative approximate MMSE estimator based on the turbo principle [16], which can be applied to any nonlinear function of the linearly mapped signal to be estimated. Furthermore, the only constraint on the distribution of the signal to be estimated is that its elements must be independent and identically distributed (i.i.d.) random variables. Therefore, the GEC-SR algorithm can be used as an approximate MMSE channel estimator for any ADC resolutions ranging from 1 bit to high-resolution ADCs. However, the inverse of the sensing matrix is required at each iteration, which is impractical in massive MIMO systems in the complexity point of view.

The generalized approximate message passing-based (GAMP-based) channel estimator for mmWave massive MIMO systems with low-resolution ADCs was proposed in [15], which is another iterative approximate MMSE channel estimator. In contrast to the GEC-SR algorithm, only matrix-vector multiplications are required at each iteration, which is favorable in the complexity point of view. As in the GEC-SR algorithm, the GAMP-based algorithm can be applied to any ADC resolutions and any channel distributions as long as the elements of channel are i.i.d. random variable. The performance of the GEC-SR and GAMP algorithms, however, cannot be guaranteed when the sensing matrix is ill-conditioned, which frequently occurs in the mmWave band. To prevent the sensing matrix from becoming ill-conditioned, the GAMP-based channel estimator constructs the virtual channel representation using discrete Fourier transform (DFT) matrices, whose columns are orthogonal. However, such virtual channel representation results in a large gridding error, which leads to performance degradation.

Our goal is to propose an iterative approximate maximum a posteriori (MAP) channel estimator for mmWave massive MIMO systems with 1 bit ADCs. Due to the sparse nature, the MAP channel estimation problem is converted to a sparsity-constrained optimization problem, which is NP-hard to solve *[17]*. To approximately solve such problems iteratively, the gradient support pursuit (GraSP) and gradient hard thresholding pursuit (GraHTP) algorithms were proposed in [17] and [18], respectively. The GraSP and GraHTP algorithms pursue the gradient of the objective function at each iteration by hard thresholding. These algorithms are the generalizations of the compressive sampling matching pursuit (CoSaMP) [19] and hard thresholding pursuit (HTP) [20] algorithms, respectively, in compressive sensing (CS).

With highly coherent sensing matrix, however, the GraSP and GraHTP algorithms do not perform appropriately since the objective function becomes ill-conditioned. To remedy such break down, we exploit the band maximum selecting (BMS) hard thresholding technique, which is then applied to the GraSP and GraHTP algorithms to propose the BMSGraSP and BMSGraHTP algorithms, respectively. The proposed BMS-based algorithms perform hard thresholding for the gradient of the objective function based on the proposed band maximum criterion, which tests whether an index is the ground truth index or the by-product of another index. To reduce the complexity of the BMS-based algorithms, a fast Fourier transform-based (FFT-based) fast implementation of the objective function and gradient is proposed. The BMS-based algorithms are shown to be both accurate and efficient, which is verified through extensive simulations.

The rest of this paper is organized as follows. In Section 2, mmWave massive MIMO systems with 1 bit ADCs are described. In Section 3, the MAP channel estimation framework is formulated. In Section 4, the BMS hard thresholding technique is proposed, which is applied to the GraSP and GraHTP algorithms. In addition, an FFT-based fast implementation is proposed. In Section 5, the results and discussion are presented, and the conclusions are followed in Section 6.

Notation: a, $\mathbf{a}$, and $\mathbf{A}$ denote a scalar, vector, and matrix, respectively. $\|\mathbf{a}\|_0$, $\|\mathbf{a}\|_1$, and $\|\mathbf{a}\|$ represent the L^0-, L^1-, and L^2-norm of $\mathbf{a}$, respectively. $\|\mathbf{A}\|_F$ is the Frobenius norm of $\mathbf{A}$. The transpose, conjugate transpose, and conjugate of $\mathbf{A}$ are denoted as $\mathbf{A}^T$, $\mathbf{A}^H$, and $\overline{\mathbf{A}}$, respectively. The element-wise vector multiplication and division of $\mathbf{a}$ and $\mathbf{b}$ are denoted as $\mathbf{a} \odot \mathbf{b}$ and $\mathbf{a} \oslash \mathbf{b}$, respectively. The sum of all of the elements of $\mathbf{a}$ is denoted as sum($\mathbf{a}$). The vectorization of $\mathbf{A}$ is denoted as vec($\mathbf{A}$), which is formed by stacking all of the columns of $\mathbf{A}$. The unvectorization of $\mathbf{a}$ is denoted as unvec($\mathbf{a}$), which is the inverse of vec($\mathbf{A}$). The Kronecker product of $\mathbf{A}$ and $\mathbf{B}$ is denoted as $\mathbf{A} \otimes \mathbf{B}$. The support of $\mathbf{a}$ is denoted as supp($\mathbf{a}$), which is the set of indices formed by collecting all of the indices of the non-zero elements of $\mathbf{a}$. The best s-term approximation of $\mathbf{a}$ is denoted as $\mathbf{a}|_s$, which is formed by leaving only the s largest (in absolute value) elements of $\mathbf{a}$ and replacing

the other elements with 0. Similarly, the vector obtained by leaving only the elements of $\mathbf{a}$ indexed by the set $\mathcal{A}$ and replacing the other elements with 0 is denoted as $\mathbf{a}|_{\mathcal{A}}$. The absolute value of a scalar a and cardinality of a set $\mathcal{A}$ are denoted as $|a|$ and $|\mathcal{A}|$, respectively. The set difference between the sets $\mathcal{A}$ and $\mathcal{B}$, namely, $\mathcal{A} \cap \mathcal{B}^c$, is denoted as $\mathcal{A} \setminus \mathcal{B}$. $\phi(\mathbf{a})$ and $\Phi(\mathbf{a})$ are element-wise standard normal PDF and CDF functions of $\mathbf{a}$, whose ith elements are $\frac{1}{\sqrt{2\pi}}e^{-\frac{a_i^2}{2}}$ and $\int_{-\infty}^{a_i} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx$, respectively. The $m \times 1$ zero vector and $m \times m$ identity matrix are denoted as $\mathbf{0}_m$ and $\mathbf{I}_m$, respectively.

2 mmWave massive MIMO systems with 1 bit ADCs

2.1 System model

As shown in Fig. 1, consider a mmWave massive MIMO system with uniform linear arrays (ULAs) at the transmitter and receiver. The N-antenna transmitter transmits a training sequence of length T to the M-antenna receiver. Therefore, the received signal $\mathbf{Y} = [\, \mathbf{y}[1] \quad \mathbf{y}[2] \quad \cdots \quad \mathbf{y}[T]\,] \in \mathbb{C}^{M \times T}$ is

$$\mathbf{Y} = \sqrt{\rho}\mathbf{H}\mathbf{S} + \mathbf{N}, \tag{1}$$

which is the collection of the t-th received signal $\mathbf{y}[t] \in \mathbb{C}^M$ over $t \in \{1, \ldots, T\}$. In the mmWave band, the channel $\mathbf{H} \in \mathbb{C}^{M \times N}$ consists of a small number of paths, whose parameters are the path gains, angle-of-arrivals (AoAs), and angle-of-departures (AoDs) [21]. Therefore, $\mathbf{H}$ is

$$\mathbf{H} = \sum_{\ell=1}^{L} \alpha_\ell \mathbf{a}_{\mathrm{RX}}(\theta_{\mathrm{RX},\ell}) \mathbf{a}_{\mathrm{TX}}(\theta_{\mathrm{TX},\ell})^{\mathrm{H}} \tag{2}$$

where L is the number of paths, $\alpha_\ell \in \mathbb{C}$ is the path gain of the ℓ-th path, and $\theta_{\mathrm{RX},\ell} \in [-\pi/2, \pi/2]$ and $\theta_{\mathrm{TX},\ell} \in [-\pi/2, \pi/2]$ are the AoA and AoD of the ℓth path, respectively. The steering vectors $\mathbf{a}_{\mathrm{RX}}(\theta_{\mathrm{RX},\ell}) \in \mathbb{C}^M$ and $\mathbf{a}_{\mathrm{TX}}(\theta_{\mathrm{TX},\ell}) \in \mathbb{C}^N$ are

$$\mathbf{a}_{\mathrm{RX}}(\theta_{\mathrm{RX},\ell}) = \frac{1}{\sqrt{M}} \left[\, 1 \,\cdots\, e^{-j\pi(M-1)\sin(\theta_{\mathrm{RX},\ell})} \,\right]^{\mathrm{T}}, \tag{3}$$

$$\mathbf{a}_{\mathrm{TX}}(\theta_{\mathrm{TX},\ell}) = \frac{1}{\sqrt{N}} \left[\, 1 \,\cdots\, e^{-j\pi(N-1)\sin(\theta_{\mathrm{TX},\ell})} \,\right]^{\mathrm{T}} \tag{4}$$

where the inter-element spacing is half-wavelength. The training sequence $\mathbf{S} = [\, \mathbf{s}[1] \quad \mathbf{s}[2] \quad \cdots \quad \mathbf{s}[T]\,] \in \mathbb{C}^{N \times T}$ is the collection of the tth training sequence $\mathbf{s}[t] \in \mathbb{C}^N$ over $t \in \{1, \ldots, T\}$, whose power constraint is $\|\mathbf{s}[t]\|^2 = N$. The additive white Gaussian noise (AWGN) $\mathbf{N} = [\, \mathbf{n}[1] \quad \mathbf{n}[2] \quad \cdots \quad \mathbf{n}[T]\,] \in \mathbb{C}^{M \times T}$ is the collection of the tth AWGN $\mathbf{n}[t] \in \mathbb{C}^M$ over $t \in \{1, \ldots, T\}$, which is distributed as $\mathrm{vec}(\mathbf{N}) \sim \mathcal{CN}(\mathbf{0}_{MT}, \mathbf{I}_{MT})$. The signal-to-noise ratio (SNR) is defined as ρ.

At the receiver, the real and imaginary parts of the received signal are quantized by 1 bit ADCs. The quantized received signal $\hat{\mathbf{Y}}$ is

$$\begin{aligned} \hat{\mathbf{Y}} &= \mathrm{Q}(\mathbf{Y}) \\ &= \mathrm{Q}(\sqrt{\rho}\mathbf{H}\mathbf{S} + \mathbf{N}) \end{aligned} \tag{5}$$

where $\mathrm{Q}(\cdot)$ is the 1 bit quantization function, whose threshold is 0. Therefore, $\mathrm{Q}(\mathbf{Y})$ is

$$\mathrm{Q}(\mathbf{Y}) = \mathrm{sign}(\mathrm{Re}(\mathbf{Y})) + j\,\mathrm{sign}(\mathrm{Im}(\mathbf{Y})) \tag{6}$$

where $\mathrm{sign}(\cdot)$ is the element-wise sign function. The goal is to estimate $\mathbf{H}$ by estimating $\{\alpha_\ell\}_{\ell=1}^{L}$, $\{\theta_{\mathrm{RX},\ell}\}_{\ell=1}^{L}$, and $\{\theta_{\mathrm{TX},\ell}\}_{\ell=1}^{L}$ from $\hat{\mathbf{Y}}$.
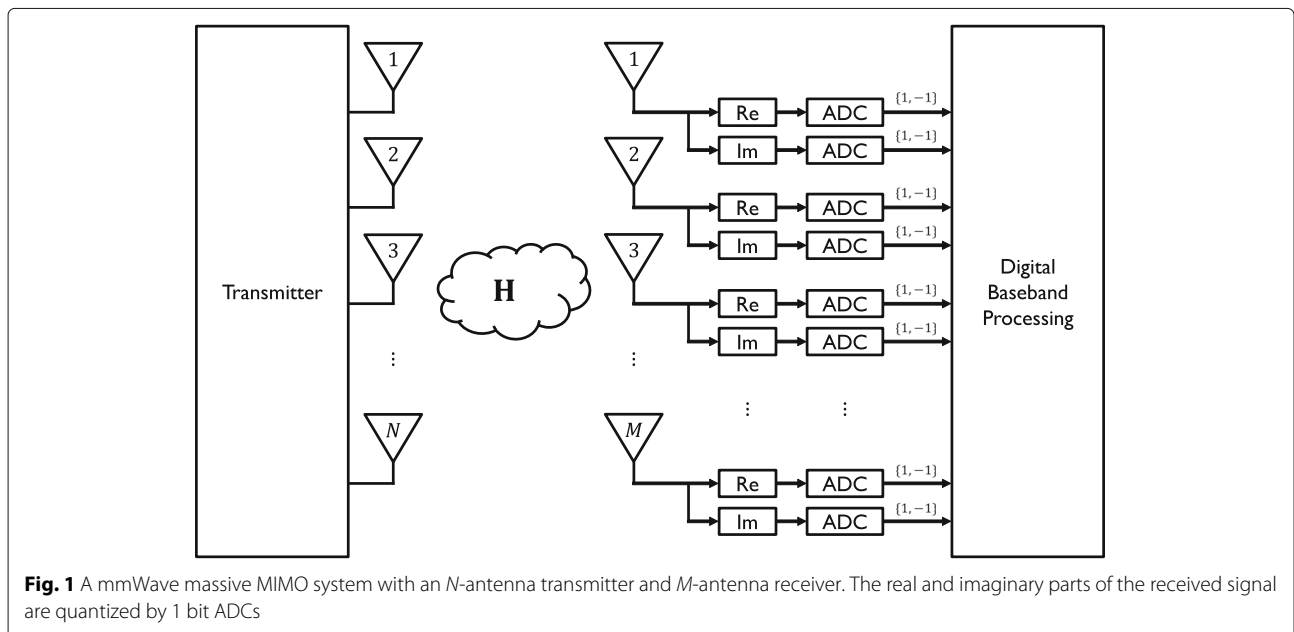

Fig. 1 A mmWave massive MIMO system with an N-antenna transmitter and M-antenna receiver. The real and imaginary parts of the received signal are quantized by 1 bit ADCs

2.2 Virtual channel representation

In the mmWave channel model in (2), $\{\theta_{\mathrm{RX},\ell}\}_{\ell=1}^{L}$ and $\{\theta_{\mathrm{TX},\ell}\}_{\ell=1}^{L}$ are hidden in $\{\mathbf{a}_{\mathrm{RX}}(\theta_{\mathrm{RX},\ell})\}_{\ell=1}^{L}$ and $\{\mathbf{a}_{\mathrm{TX}}(\theta_{\mathrm{TX},\ell})\}_{\ell=1}^{L}$, respectively. The non-linear mapping of $\{\theta_{\mathrm{RX},\ell}\}_{\ell=1}^{L}$ and $\{\theta_{\mathrm{TX},\ell}\}_{\ell=1}^{L}$ to $\mathbf{Y}$ renders a non-linear channel estimation problem. To convert the non-linear channel estimation problem to a linear channel estimation problem, we adopt the virtual channel representation [22].

The virtual channel representation of $\mathbf{H}$ is

$$\mathbf{H} \approx \mathbf{A}_{\mathrm{RX}}\mathbf{X}^{*}\mathbf{A}_{\mathrm{TX}}^{\mathrm{H}} \tag{7}$$

where the dictionary pair $\mathbf{A}_{\mathrm{RX}} \in \mathbb{C}^{M \times B_{\mathrm{RX}}}$ and $\mathbf{A}_{\mathrm{TX}} \in \mathbb{C}^{N \times B_{\mathrm{TX}}}$ is the collection of $B_{\mathrm{RX}} \geq M$ steering vectors and $B_{\mathrm{TX}} \geq N$ steering vectors, respectively. Therefore, $\mathbf{A}_{\mathrm{RX}}$ and $\mathbf{A}_{\mathrm{TX}}$ are

$$\mathbf{A}_{\mathrm{RX}} = \left[\ \mathbf{a}_{\mathrm{RX}}(\omega_{\mathrm{RX},1})\ \cdots\ \mathbf{a}_{\mathrm{RX}}(\omega_{\mathrm{RX},B_{\mathrm{RX}}})\ \right], \tag{8}$$

$$\mathbf{A}_{\mathrm{TX}} = \left[\ \mathbf{a}_{\mathrm{TX}}(\omega_{\mathrm{TX},1})\ \cdots\ \mathbf{a}_{\mathrm{TX}}(\omega_{\mathrm{TX},B_{\mathrm{TX}}})\ \right], \tag{9}$$

whose gridding AoAs $\{\omega_{\mathrm{RX},i}\}_{i=1}^{B_{\mathrm{RX}}}$ and AoDs $\{\omega_{\mathrm{TX},j}\}_{j=1}^{B_{\mathrm{TX}}}$ are selected so as to form overcomplete DFT matrices. The gridding AoAs and AoDs are the B_{RX} and B_{TX} points from $[-\pi/2, \pi/2]$, respectively, to discretize the AoAs and AoDs because the ground truth AoAs and AoDs are unknown. To make a dictionary pair of the overcomplete DFT matrix form, the gridding AoAs and AoDs are given as $\omega_{\mathrm{RX},i} = \sin^{-1}(-1 + 2/B_{\mathrm{RX}} \cdot (i-1))$ and $\omega_{\mathrm{RX},j} = \sin^{-1}(-1 + 2/B_{\mathrm{TX}} \cdot (j-1))$, respectively. We prefer overcomplete DFT matrices because they are relatively well-conditioned, and DFT matrices are friendly to the FFT-based implementation, which will be discussed in Section 4. The virtual channel $\mathbf{X}^{*} \in \mathbb{C}^{B_{\mathrm{RX}} \times B_{\mathrm{TX}}}$ is the collection of $\{\alpha_{\ell}\}_{\ell=1}^{L}$, whose (i,j)th element is α_{ℓ} whenever $(\omega_{\mathrm{RX},i}, \omega_{\mathrm{TX},j})$ is the nearest to $(\theta_{\mathrm{RX},\ell}, \theta_{\mathrm{TX},\ell})$ but 0 otherwise. In general, the error between $\mathbf{H}$ and $\mathbf{A}_{\mathrm{RX}}\mathbf{X}^{*}\mathbf{A}_{\mathrm{TX}}^{\mathrm{H}}$ is inversely proportional to B_{RX} and B_{TX}. To approximate $\mathbf{H}$ using (7) with negligible error, the dictionary pair must be dense, namely, $B_{\mathrm{RX}} \gg M$ and $B_{\mathrm{TX}} \gg N$.

Before we proceed, we provide a supplementary explanation on the approximation in (7). In this work, we attempt to estimate the L-sparse $\mathbf{X}^{*}$ in (7) because the sparse assumption on $\mathbf{X}^{*}$ is favorable when the goal is to formulate the channel estimation problem as a sparsity-constrained problem. The cost of assuming that $\mathbf{X}^{*}$ is L-sparse is, as evident, the approximation error shown in (7). Alternatively, the approximation error can be perfectly removed by considering $\mathbf{X}^{*}$ satisfying $\mathbf{H} = \mathbf{A}_{\mathrm{RX}}\mathbf{X}^{*}\mathbf{A}_{\mathrm{TX}}^{\mathrm{H}}$, i.e., equality instead of approximation. One well-known $\mathbf{X}^{*}$ satisfying the equality is the minimum Frobenius norm solution, i.e., $\mathbf{X}^{*} = \mathbf{A}_{\mathrm{RX}}^{\mathrm{H}}(\mathbf{A}_{\mathrm{RX}}\mathbf{A}_{\mathrm{RX}}^{\mathrm{H}})^{-1}\mathbf{H}(\mathbf{A}_{\mathrm{TX}}\mathbf{A}_{\mathrm{TX}}^{\mathrm{H}})^{-1}\mathbf{A}_{\mathrm{TX}}$. Such $\mathbf{X}^{*}$, however,

has no evident structure to exploit in channel estimation, which is the reason why we assume that $\mathbf{X}^{*}$ is L-sparse at the cost of the approximation error in (7).

In practice, the arrays at the transmitter and receiver are typically large to compensate the path loss in the mmWave band, whereas the number of line-of-sight (LOS) and near LOS paths is small [23]. Therefore, $\mathbf{X}^{*}$ is sparse when the dictionary pair is dense because only L elements among $B_{\mathrm{RX}}B_{\mathrm{TX}}$ elements are non-zero where $L \ll MN \ll B_{\mathrm{RX}}B_{\mathrm{TX}}$. In the sequel, we use the shorthand notation $B = B_{\mathrm{RX}}B_{\mathrm{TX}}$.

To facilitate the channel estimation framework, we vectorize (1) and (5) in conjunction with (7). First, note that

$$\mathbf{Y} = \sqrt{\rho}\mathbf{A}_{\mathrm{RX}}\mathbf{X}^{*}\mathbf{A}_{\mathrm{TX}}^{\mathrm{H}}\mathbf{S} + \mathbf{N} + \mathbf{E} \tag{10}$$

where the gridding error $\mathbf{E} \in \mathbb{C}^{M \times T}$ represents the mismatch in (7).[1] Then, the vectorized received signal $\mathbf{y} = \mathrm{vec}(\mathbf{Y}) \in \mathbb{C}^{MT}$ is

$$\mathbf{y} = \sqrt{\rho}\mathbf{A}\mathbf{x}^{*} + \mathbf{n} + \mathbf{e} \tag{11}$$

where

$$\begin{aligned}\mathbf{A} &= \mathbf{S}^{\mathrm{T}}\overline{\mathbf{A}}_{\mathrm{TX}} \otimes \mathbf{A}_{\mathrm{RX}} \\ &= \left[\ \mathbf{a}_1\ \mathbf{a}_2\ \cdots\ \mathbf{a}_B\ \right],\end{aligned} \tag{12}$$

$$\begin{aligned}\mathbf{x}^{*} &= \mathrm{vec}(\mathbf{X}^{*}) \\ &= \left[\ x_1^{*}\ x_2^{*}\ \cdots\ x_B^{*}\ \right]^{\mathrm{T}},\end{aligned} \tag{13}$$

$$\mathbf{n} = \mathrm{vec}(\mathbf{N}), \tag{14}$$

$$\mathbf{e} = \mathrm{vec}(\mathbf{E}). \tag{15}$$

The vectorized quantized received signal $\hat{\mathbf{y}} = \mathrm{vec}(\hat{\mathbf{Y}}) \in \mathbb{C}^{MT}$ is

$$\begin{aligned}\hat{\mathbf{y}} &= Q(\mathbf{y}) \\ &= Q(\sqrt{\rho}\mathbf{A}\mathbf{x}^{*} + \mathbf{n} + \mathbf{e}).\end{aligned} \tag{16}$$

The goal is to estimate L-sparse $\mathbf{x}^{*}$ from $\hat{\mathbf{y}}$.

3 Problem formulation

In this section, we formulate the channel estimation problem using the MAP criterion. To facilitate the MAP channel estimation framework, the real counterparts of the complex forms in (16) are introduced. Then, the likelihood function of $\mathbf{x}^{*}$ is derived.

The real counterparts are the collections of the real and imaginary parts of the complex forms. Therefore, the real counterparts $\hat{\mathbf{y}}_{\mathrm{R}} \in \mathbb{R}^{2MT}$, $\mathbf{A}_{\mathrm{R}} \in \mathbb{R}^{2MT \times 2B}$, and $\mathbf{x}_{\mathrm{R}}^{*} \in \mathbb{R}^{2B}$ are

---

[1]In practice, $\mathbf{X}^{*}$ may be either approximately sparse or exactly sparse to formulate (10). If $\mathbf{X}^{*}$ is approximately sparse, the leakage effect is taken into account so the mismatch in (7) becomes 0, namely, $\mathrm{vec}(\mathbf{E}) = \mathbf{0}_{MT}$. In contrast, the mismatch in (7) must be taken into account with a non-zero $\mathbf{E}$ when $\mathbf{X}^{*}$ is exactly sparse. Fortunately, $\mathbf{E}$ is inversely proportional to B_{RX} and B_{TX}. Therefore, we adopt the latter definition of $\mathbf{X}^{*}$ and propose our algorithm ignoring $\mathbf{E}$ assuming that $B_{\mathrm{RX}} \gg M$ and $B_{\mathrm{TX}} \gg N$. The performance degradation from $\mathbf{E}$ will become less as B_{RX} and B_{TX} become sufficiently large.

$$\hat{\mathbf{y}}_R = \begin{bmatrix} \mathrm{Re}(\hat{\mathbf{y}})^T & \mathrm{Im}(\hat{\mathbf{y}})^T \end{bmatrix}^T$$
$$= \begin{bmatrix} \hat{y}_{R,1} & \hat{y}_{R,2} & \cdots & \hat{y}_{R,2MT} \end{bmatrix}^T, \tag{17}$$

$$\mathbf{A}_R = \begin{bmatrix} \mathrm{Re}(\mathbf{A}) & -\mathrm{Im}(\mathbf{A}) \\ \mathrm{Im}(\mathbf{A}) & \mathrm{Re}(\mathbf{A}) \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{a}_{R,1} & \mathbf{a}_{R,2} & \cdots & \mathbf{a}_{R,2MT} \end{bmatrix}^T, \tag{18}$$

$$\mathbf{x}_R^* = \begin{bmatrix} \mathrm{Re}(\mathbf{x}^*)^T & \mathrm{Im}(\mathbf{x}^*)^T \end{bmatrix}^T$$
$$= \begin{bmatrix} x_{R,1}^* & x_{R,2}^* & \cdots & x_{R,2B}^* \end{bmatrix}^T, \tag{19}$$

which are the collections of the real and imaginary parts of $\hat{\mathbf{y}}$, $\mathbf{A}$, and $\mathbf{x}^*$, respectively. In the sequel, we use the complex forms and the real counterparts interchangeably. For example, $\mathbf{x}^*$ and $\mathbf{x}_R^*$ refer to the same entity.

Before we formulate the likelihood function of $\mathbf{x}^*$, note that $\mathbf{e}$ is hard to analyze. However, $\mathbf{e}$ is negligible when the dictionary pair is dense. Therefore, we formulate the likelihood function of $\mathbf{x}^*$ without $\mathbf{e}$. The price of such oversimplification is negligible when $B_{RX} \gg M$ and $B_{TX} \gg N$, which is to be shown in Section 5 where $\mathbf{e} \neq \mathbf{0}_{MT}$. To derive the likelihood function of $\mathbf{x}^*$, note that

$$\sqrt{\rho}\mathbf{A}\mathbf{x}^* + \mathbf{n} \sim \mathcal{CN}(\sqrt{\rho}\mathbf{A}\mathbf{x}^*, \mathbf{I}_{MT}) \tag{20}$$

given $\mathbf{x}^*$. Then, from (20) in conjunction with (16), the log-likelihood function $f(\mathbf{x})$ is [10]

$$f(\mathbf{x}) = \log \mathrm{Pr}\left[\hat{\mathbf{y}} = \mathrm{Q}(\sqrt{\rho}\mathbf{A}\mathbf{x} + \mathbf{n}) \mid \mathbf{x} \right]$$
$$= \sum_{i=1}^{2MT} \log \Phi\left(\sqrt{2\rho}\hat{y}_{R,i}\mathbf{a}_{R,i}^T\mathbf{x}_R \right). \tag{21}$$

If the distribution of $\mathbf{x}^*$ is known, the MAP estimate of $\mathbf{x}^*$ is

$$\underset{\mathbf{x} \in \mathbb{C}^B}{\mathrm{argmax}} \, (f(\mathbf{x}) + g_{MAP}(\mathbf{x})) \tag{22}$$

where $g_{MAP}(\mathbf{x})$ is the logarithm of the PDF of $\mathbf{x}^*$. In practice, however, $g_{MAP}(\mathbf{x})$ is unknown. Therefore, we formulate the MAP channel estimation framework based on $\{\alpha_\ell\}_{\ell=1}^L$, $\{\theta_{RX,\ell}\}_{\ell=1}^L$, and $\{\theta_{TX,\ell}\}_{\ell=1}^L$ where we assume the followings:

1. $\alpha_\ell \sim \mathcal{CN}(0,1)$ for all ℓ
2. $\theta_{RX,\ell} \sim \mathrm{unif}([-\pi/2, \pi/2])$ for all ℓ
3. $\theta_{TX,\ell} \sim \mathrm{unif}([-\pi/2, \pi/2])$ for all ℓ
4. $\{\alpha_\ell\}_{\ell=1}^L$, $\{\theta_{RX,\ell}\}_{\ell=1}^L$, and $\{\theta_{TX,\ell}\}_{\ell=1}^L$ are independent.

Then, the MAP estimate of $\mathbf{x}^*$ considering the channel sparsity is

$$\underset{\mathbf{x} \in \mathbb{C}^B}{\mathrm{argmax}} \, (f(\mathbf{x}) + g(\mathbf{x})) \quad \text{s.t.} \quad \|\mathbf{x}\|_0 \le L \tag{23}$$

where $g(\mathbf{x}) = -\|\mathbf{x}_R\|^2$ is the logarithm of the PDF of $\mathcal{CN}(\mathbf{0}_B, \mathbf{I}_B)$ ignoring the constant factor. However, note that only the optimization problems (22) and (23) are equivalent in the sense that their solutions are the same,

not $g_{MAP}(\mathbf{x})$ and $g(\mathbf{x})$. In the ML channel estimation framework, the ML estimate of $\mathbf{x}^*$ is

$$\underset{\mathbf{x} \in \mathbb{C}^B}{\mathrm{argmax}} \, f(\mathbf{x}) \quad \text{s.t.} \quad \|\mathbf{x}\|_0 \le L. \tag{24}$$

In the sequel, we focus on solving (23) because (23) reduces to (24) when $g(\mathbf{x}) = 0$. In addition, we denote the objective function and the gradient in (23) as $h(\mathbf{x})$ and $\nabla h(\mathbf{x})$, respectively. Therefore,

$$h(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x}), \tag{25}$$
$$\nabla h(\mathbf{x}) = \nabla f(\mathbf{x}) + \nabla g(\mathbf{x})$$
$$= \begin{bmatrix} \nabla h(x_1) & \nabla h(x_2) & \cdots & \nabla h(x_B) \end{bmatrix}^T \tag{26}$$

where the differentiation is with respect to $\mathbf{x}$.

4 Channel estimation via gradient pursuit
In this section, we propose the BMSGraSP and BMSGraHTP algorithms to solve (23), which are the variants of the GraSP [17] and GraHTP [18] algorithms, respectively. Then, an FFT-based fast implementation is proposed. In addition, we investigate the limit of the BMSGraSP and BMSGraHTP algorithms in the high SNR regime in 1 bit ADCs.

4.1 Proposed BMSGraSP and BMSGraHTP algorithms
Note that $h(\mathbf{x})$ in (23) is concave because $f(\mathbf{x})$ and $g(\mathbf{x})$ are the sums of the logarithms of $\Phi(\cdot)$ and $\phi(\cdot)$, respectively, which are log-concave [24]. However, (23) is not a convex optimization problem because the sparsity constraint is not convex. Furthermore, solving (23) is NP-hard because of its combinatorial complexity. To approximately optimize convex objective functions with sparsity constraints iteratively by pursuing the gradient of the objective function, the GraSP and GraHTP algorithms were proposed in [17] and [18], respectively.

To solve (23), the GraSP and GraHTP algorithms roughly proceed as follows at each iteration when $\mathbf{x}$ is the current estimate of $\mathbf{x}^*$ where the iteration index is omitted for simplicity. First, the best L-term approximation of $\nabla h(\mathbf{x})$ is computed, which is

$$T_L(\nabla h(\mathbf{x})) = \nabla h(\mathbf{x})|_L \tag{27}$$

where $T_L(\cdot)$ is the L-term hard thresholding function. Here, $T_L(\cdot)$ leaves only the L largest elements (in absolute value) of $\nabla h(\mathbf{x})$, and sets all the other remaining elements to 0. Then, after the estimate of $\mathrm{supp}(\mathbf{x}^*)$ is updated by selecting

$$\mathcal{I} = \mathrm{supp}(T_L(\nabla h(\mathbf{x}))), \tag{28}$$

i.e., $\mathcal{I}$ is the set of indices formed by collecting the L indices of $\nabla h(\mathbf{x})$ corresponding to its L largest elements

(in absolute value), the estimate of $\mathbf{x}^*$ is updated by solving the following optimization problem

$$\underset{\mathbf{x}\in\mathbb{C}^B}{\arg\max}\ h(\mathbf{x})\ \text{s.t.}\ \text{supp}(\mathbf{x})\subseteq\mathcal{I}, \tag{29}$$

which can be solved using convex optimization because the support constraint is convex [24]. The GraSP and GraHTP algorithms are the generalizations of the CoSaMP [19] and HTP [20] algorithms, respectively. This follows because the gradient of the squared error is the scaled proxy of the residual.

To solve (23) using the GraSP and GraHTP algorithms, $h(\mathbf{x})$ is required either to have a stable restricted Hessian [17] or to be strongly convex and smooth [18]. These conditions are the generalizations of the restricted isometry property (RIP) in CS [25], which means that $h(\mathbf{x})$ is likely to satisfy these conditions when $\mathbf{A}$ is either a restricted isometry, well-conditioned, or incoherent. In practice, however, $\mathbf{A}$ is highly coherent because the dictionary pair is typically dense to reduce the mismatch in (7).

To illustrate how the GraSP and GraHTP algorithms fail to solve (23) when $\mathbf{A}$ is highly coherent, consider the real counterpart of $\nabla h(\mathbf{x})$. The real counterpart $\nabla h(\mathbf{x}_R) \in \mathbb{R}^{2B}$ is

$$\begin{aligned}
&\nabla h(\mathbf{x}_R)\\
&= \begin{bmatrix} \text{Re}(\nabla h(\mathbf{x}))^T & \text{Im}(\nabla h(\mathbf{x}))^T \end{bmatrix}^T\\
&= \sum_{i=1}^{2MT} \lambda\left(\sqrt{2\rho}\,\hat{y}_{R,i}\mathbf{a}_{R,i}^T\mathbf{x}_R\right)\sqrt{2\rho}\,\hat{y}_{R,i}\mathbf{a}_{R,i} - 2\mathbf{x}_R,
\end{aligned} \tag{30}$$

which follows from $\nabla\log\Phi(\mathbf{a}_R^T\mathbf{x}_R) = \lambda(\mathbf{a}_R^T\mathbf{x}_R)\mathbf{a}_R$ and $\nabla\|\mathbf{x}_R\|^2 = 2\mathbf{x}_R$ where $\lambda(\cdot) = \phi(\cdot)\oslash\Phi(\cdot)$ is the inverse Mills ratio function[2]. Then, the following observation holds from directly computing $\nabla h(x_i)$, whose real and imaginary parts are the i-th and $(i+B)$-th elements of $\nabla h(\mathbf{x}_R)$, respectively.

Observation 1 $\nabla h(x_i) = \nabla h(x_j)$ *if* $\mathbf{a}_i = \mathbf{a}_j$ *and* $x_i = x_j$.

However, Observation 1 is meaningless because $\mathbf{a}_i \neq \mathbf{a}_j$ unless $i = j$. To establish a meaningful observation, consider the coherence between $\mathbf{a}_i$ and $\mathbf{a}_j$, which reflects the proximity between $\mathbf{a}_i$ and $\mathbf{a}_j$ according to [26, 27]

$$\mu(i,j) = \frac{|\mathbf{a}_i^H\mathbf{a}_j|}{\|\mathbf{a}_i\|\|\mathbf{a}_j\|}. \tag{31}$$

Then, using the η-coherence band, which is [26]

$$B_\eta(i) = \{j \mid \mu(i,j) \geq \eta\} \tag{32}$$

where $\eta \in (0,1)$, we establish the following conjecture when η is sufficiently large.

Conjecture 1 $\nabla h(x_i) \approx \nabla h(x_j)$ *if* $j \in B_\eta(i)$ *and* $x_i = x_j$.

At this point, we use Conjecture 1 to illustrate how the GraSP and GraHTP algorithms fail to estimate $\text{supp}(\mathbf{x}^*)$ from (28) by naive hard thresholding when $\mathbf{A}$ is highly coherent. To proceed, consider the following example, which assumes that $\mathbf{x}^*$ and $\hat{\mathbf{y}}$ are realized with $\mathbf{x}$ representing the current estimate of $\mathbf{x}^*$ so as to satisfy

1) $i = \underset{k\in\{1,2,\dots,B\}}{\arg\max} |x_k^*|$
2) $i = \underset{k\in\{1,2,\dots,B\}}{\arg\max} |\nabla h(x_k)|$
3) $\mathcal{J}\cap\text{supp}(\mathbf{x}^*) = \emptyset$

where i is the index corresponding to the largest element of the ground truth[3] virtual channel $\mathbf{x}^*$, and

$$\mathcal{J} = \{j \mid j \in B_\eta(i), x_i = x_j\}\setminus\{i\} \tag{33}$$

is the by-product of i. Here, $\mathcal{J}$ is called the by-product of i because

$$\begin{aligned}
|\nabla h(x_j)| &\approx |\nabla h(x_i)|\\
&= \max_{k\in\{1,2,\dots,B\}}|\nabla h(x_k)|,
\end{aligned} \tag{34}$$

which follows from Conjecture 1, holds despite $x_j^* = 0$ for all $j \in \mathcal{J}$. In other words, the by-product of i refers to the fact that $\nabla h(x_i)$ and $\nabla h(x_j)$ are indistinguishable for all $j \in \mathcal{J}$ according to (34), but the elements of $\mathbf{x}^*$ indexed by $\mathcal{J}$ are 0 according to 3). Therefore, when we attempt to estimate $\text{supp}(\mathbf{x}^*)$ by hard thresholding $\nabla h(\mathbf{x})$, the indices in $\mathcal{J}$ will likely be erroneously selected as the estimate of $\text{supp}(\mathbf{x}^*)$ because $\nabla h(x_j)$ and the maximum element of $\nabla h(\mathbf{x})$, which is $\nabla h(x_i)$ according to 2), are indistinguishable for all $j \in \mathcal{J}$.

To illustrate how (28) cannot estimate $\text{supp}(\mathbf{x}^*)$ when $\mathbf{A}$ is highly coherent, consider another example where $\nabla h(\mathbf{x})$ and $T_L(\nabla h(\mathbf{x}))$ are shown in Figs. 2 and 3, respectively. In this example, $\text{supp}(\mathbf{x}^*)$ is widely spread, whereas most of $\text{supp}(T_L(\nabla h(\mathbf{x})))$ are in the coherence band of the index of the maximum element of $\nabla h(\mathbf{x})$. This shows that hard thresholding $\nabla h(\mathbf{x})$ is not sufficient to distinguish whether an index is the ground truth index or the by-product of another index. To solve this problem, we propose the BMS hard thresholding technique.

The BMS hard thresholding function $T_{\text{BMS},L}(\cdot)$ is an L-term hard thresholding function, which is proposed based on Conjecture 1. The BMS hard thresholding technique is presented in Algorithm 1. Line 3 selects the index of the

---

[2]The element-wise vector division in the inverse Mills ratio function is meaningless because the arguments of the inverse Mills ratio function are scalars in (30). The reason we use the element-wise vector division in the inverse Mills ratio function will become clear in (37), whose arguments are vectors.

[3]We use the term "ground truth" to emphasize that the ground truth $\mathbf{x}^*$ is the true virtual channel which actually gives the quantized received signal $\hat{\mathbf{y}}$ from (16), whereas $\mathbf{x}$ merely represents the point where $\nabla h(\mathbf{x})$ is computed to estimate $\text{supp}(\mathbf{x}^*)$ via hard thresholding.
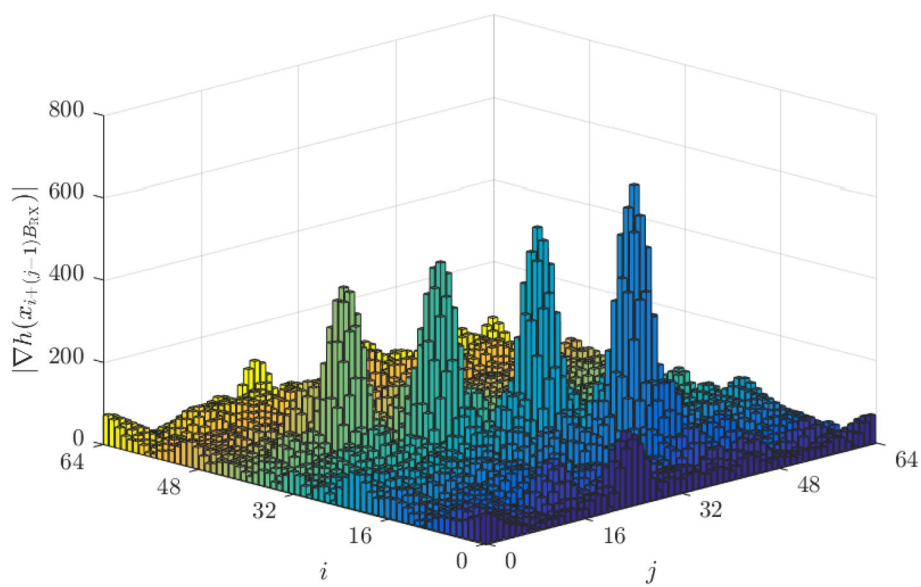
Fig. 2 The magnitude of unvec($\nabla h(\mathbf{x})$) $\in \mathbb{C}^{B_{\mathrm{RX}} \times B_{\mathrm{TX}}}$ at $\mathbf{x} = \mathbf{0}_B$, namely, before hard thresholding

maximum element of $\nabla h(\mathbf{x})$ from the unchecked index set as the current index. Line 4 constructs the by-product testing set. Line 5 checks whether the current index is greater than the by-product testing set. In this paper, Line 5 is referred to as the band maximum criterion. If the band maximum criterion is satisfied, the current index is selected as the estimate of supp($\mathbf{x}^*$) in Line 6. Otherwise, the current index is not selected as the estimate of

supp($\mathbf{x}^*$) because the current index is likely to be the by-product of another index rather than the ground truth index. Line 8 updates the unchecked index set.

Note that Algorithm 1 is a hard thresholding technique applied to $\nabla h(\mathbf{x})$. If the BMS hard thresholding technique is applied to $\mathbf{x} + \kappa \nabla h(\mathbf{x})$ where κ is the step size, $\nabla h(\mathbf{x})$ is replaced by $\mathbf{x} + \kappa \nabla h(\mathbf{x})$ in the input, output, and Lines 3, 5, and 10 of Algorithm 1. This can be derived using the
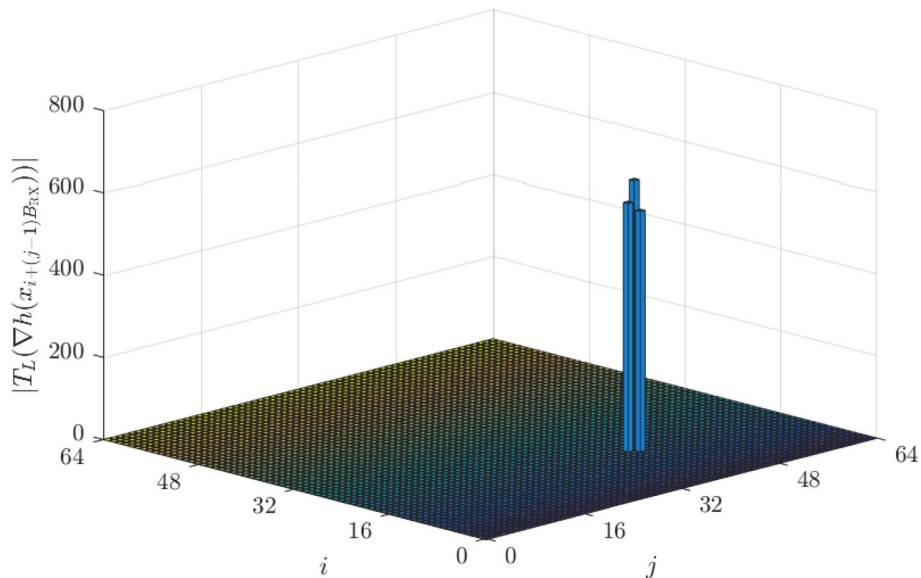

Fig. 3 The magnitude of unvec($T_L(\nabla h(\mathbf{x}))$) $\in \mathbb{C}^{B_{\mathrm{RX}} \times B_{\mathrm{TX}}}$ at $\mathbf{x} = \mathbf{0}_B$, namely, after hard thresholding. This shows how hard thresholding on $\nabla h(\mathbf{x})$ results in an incorrect estimate of supp($\mathbf{x}^*$) when $\mathbf{A}$ is highly coherent. In this example, $M = N = 16$, $B_{\mathrm{RX}} = B_{\mathrm{TX}} = 64$, $T = 20$, $L = 4$, SNR = 20 dB, and supp($\mathbf{x}^*$) is widely spread

---

Algorithm 1 BMS hard thresholding technique

Input: $\mathbf{x}$, $\nabla h(\mathbf{x})$, L
Output: $T_{\text{BMS},L}(\nabla h(\mathbf{x}))$
1: $\mathcal{S} = \emptyset, \mathcal{I} = \{1, 2, \ldots, B\}$
2: **while** $|\mathcal{S}| < L$ **do**
3:     $i = \underset{j \in \mathcal{I}}{\text{argmax}} \; |\nabla h(x_j)|$
4:     $\mathcal{J} = \{j \mid j \in B_\eta(i), x_i = x_j\} \setminus \{i\}$
5:     **if** $|\nabla h(x_i)| > \underset{j \in \mathcal{J}}{\max} |\nabla h(x_j)|$ **then**
6:         $\mathcal{S} = \mathcal{S} \cup \{i\}$
7:     **end if**
8:     $\mathcal{I} = \mathcal{I} \setminus \{i\}$
9: **end while**
10: $T_{\text{BMS},L}(\nabla h(\mathbf{x})) = \nabla h(\mathbf{x})|_{\mathcal{S}}$

---

same logic based on Conjecture 1. Now, we propose the BMSGraSP and BMSGraHTP algorithms to solve (23).

The BMSGraSP and BMSGraHTP algorithms are the variants of the GraSP and GraHTP algorithms, respectively. The difference between the BMS-based and non-BMS-based algorithms is that the hard thresholding function is $T_{\text{BMS},L}(\cdot)$ instead of $T_L(\cdot)$. The BMSGraSP and BMSGraHTP algorithms are presented in Algorithms 2 and 3, respectively. Lines 3, 4, and 5 of Algorithms 2 and 3 roughly proceed based on the same logic. Line 3 computes the gradient of the objective function. Line 4 selects $\mathcal{I}$ from the support of the hard thresholded gradient of the objective function. Line 5 maximizes the objective function subject to the support constraint. This can be solved using convex optimization because the objective function and support constraint are concave and convex, respectively. In addition, $\mathbf{b}$ is hard thresholded in Line 6 of Algorithm 2 because $\mathbf{b}$ is at most $3L$-sparse. A natural halting condition of Algorithms 2 and 3 is to halt when the current and previous $\text{supp}(\tilde{\mathbf{x}})$ are the same. The readers who are interested in a more in-depth analyses of the GraSP and GraHTP algorithms are referred to [17] and [18], respectively.

Algorithm 2 BMSGraSP algorithm

Input: $h(\cdot)$, L
Output: $\tilde{\mathbf{x}}$
1: $\tilde{\mathbf{x}} = \mathbf{0}_B$
2: **while** halting condition **do**
3:     $\mathbf{z} = \nabla h(\tilde{\mathbf{x}})$
4:     $\mathcal{I} = \text{supp}(T_{\text{BMS},2L}(\mathbf{z})) \cup \text{supp}(\tilde{\mathbf{x}})$
5:     $\mathbf{b} = \underset{\mathbf{x} \in \mathbb{C}^B}{\text{argmax}} \; h(\mathbf{x}) \text{ s.t. } \text{supp}(\mathbf{x}) \subseteq \mathcal{I}$
6:     $\tilde{\mathbf{x}} = T_L(\mathbf{b})$
7: **end while**

Algorithm 3 BMSGraHTP algorithm

Input: $h(\cdot)$, L
Output: $\tilde{\mathbf{x}}$
1: $\tilde{\mathbf{x}} = \mathbf{0}_B$
2: **while** halting condition **do**
3:     $\mathbf{z} = \tilde{\mathbf{x}} + \kappa \nabla h(\tilde{\mathbf{x}})$
4:     $\mathcal{I} = \text{supp}(T_{\text{BMS},L}(\mathbf{z}))$
5:     $\tilde{\mathbf{x}} = \underset{\mathbf{x} \in \mathbb{C}^B}{\text{argmax}} \; h(\mathbf{x}) \text{ s.t. } \text{supp}(\mathbf{x}) \subseteq \mathcal{I}$
6: **end while**

Remark 1 *Instead of hard thresholding* $\mathbf{b}$, *we can solve*

$$\tilde{\mathbf{x}} = \underset{\mathbf{x} \in \mathbb{C}^B}{argmax} \; h(\mathbf{x}) \; \text{s.t.} \; supp(\mathbf{x}) \subseteq supp(T_L(\mathbf{b})), \quad (35)$$

which is a convex optimization problem, to obtain $\tilde{\mathbf{x}}$ *in Line 6 of Algorithm 2. This is the debiasing variant of Algorithm 2* [17]. *The advantage of the debiasing variant of Algorithm 2 is a more accurate estimate of* $\mathbf{x}^*$. *However, the complexity is increased, which is incurred by solving* (35).

Remark 2 *In this paper, we assume that only* $h(\mathbf{x})$ *and* $\nabla h(\mathbf{x})$ *are required at each iteration to solve* (23) *using Algorithms 2 and 3, which can be accomplished when the first order method is used to solve convex optimization problems in Line 5 of Algorithms 2 and 3. An example of such first order method is the gradient descent method with the backtracking line search* [24].

4.2 Fast implementation via FFT

In practice, the complexity of Algorithms 2 and 3 is demanding because $h(\mathbf{x})$ and $\nabla h(\mathbf{x})$ are required at each iteration, which are high-dimensional functions defined on $\mathbb{C}^B$ where $B \gg MN$. In recent works on channel estimation and data detection in the mmWave band [14, 15, 28], the FFT-based implementation is widely used because $\mathbf{H}$ can be approximated by (7) using overcomplete DFT matrices. In this paper, an FFT-based fast implementation of $h(\mathbf{x})$ and $\nabla h(\mathbf{x})$ is proposed, which is motivated by [14, 15, 28].

To facilitate the analysis, we convert the summations in $h(\mathbf{x})$ and $\nabla h(\mathbf{x}_R)$ to matrix-vector multiplications by algebraically manipulating (21) and (30). Then, we obtain

$$h(\mathbf{x})$$
$$= \text{sum}(\log \Phi(\sqrt{2\rho}\hat{\mathbf{y}}_R \odot \mathbf{A}_R \mathbf{x}_R)) - \|\mathbf{x}_R\|^2, \quad (36)$$
$$\nabla h(\mathbf{x}_R)$$
$$= \mathbf{A}_R^{\text{T}}(\lambda(\sqrt{2\rho}\hat{\mathbf{y}}_R \odot \mathbf{A}_R \mathbf{x}_R) \odot \sqrt{2\rho}\hat{\mathbf{y}}_R) - 2\mathbf{x}_R \quad (37)$$

where we see that the bottlenecks of $h(\mathbf{x})$ and $\nabla h(\mathbf{x})$ come from the matrix-vector multiplications involving $\mathbf{A}_R$ and $\mathbf{A}_R^{\text{T}}$ resulting from the large size of $\mathbf{A}$. For example, the size of $\mathbf{A}$ is 5120×65536 in Section 5 where $M = N = 64$, $B_{\text{RX}} = B_{\text{TX}} = 256$, and $T = 80$.

To develop an FFT-based fast implementation of the matrix-vector multiplications involving $\mathbf{A}_R$ and $\mathbf{A}_R^T$, define $\mathbf{c}_R \in \mathbb{R}^{2MT}$ as $\mathbf{c}_R = \lambda(\sqrt{2\rho}\hat{\mathbf{y}}_R \odot \mathbf{A}_R \mathbf{x}_R) \odot \sqrt{2\rho}\hat{\mathbf{y}}_R$ from (37) with $\mathbf{c} \in \mathbb{C}^{MT}$ being the complex form of $\mathbf{c}_R$. From the fact that

$$\mathbf{A}_R \mathbf{x}_R = \left[\text{Re}(\mathbf{A}\mathbf{x})^T \ \text{Im}(\mathbf{A}\mathbf{x})^T \right]^T, \quad (38)$$

$$\mathbf{A}_R^T \mathbf{c}_R = \left[\text{Re}(\mathbf{A}^H \mathbf{c})^T \ \text{Im}(\mathbf{A}^H \mathbf{c})^T \right]^T, \quad (39)$$

we now attempt to compute $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^H \mathbf{c}$ via the FFT. Then, $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^H \mathbf{c}$ are unvectorized according to

$$\begin{aligned}
\text{unvec}(\mathbf{A}\mathbf{x}) &= \mathbf{A}_{RX} \mathbf{X} \mathbf{A}_{TX}^H \mathbf{S} \\
&= \mathbf{A}_{RX}(\mathbf{S}^H \underbrace{(\mathbf{A}_{TX} \mathbf{X}^H)}_{\text{FFT}})^H, \quad (40)
\end{aligned}$$

$$\begin{aligned}
\text{unvec}(\mathbf{A}^H \mathbf{c}) &= \mathbf{A}_{RX}^H \mathbf{C} \mathbf{S}^H \mathbf{A}_{TX} \\
&= \mathbf{A}_{RX}^H (\mathbf{A}_{TX}^H \underbrace{(\mathbf{S}\mathbf{C}^H)}_{\text{FFT}})^H \quad (41)
\end{aligned}$$

where $\mathbf{X} = \text{unvec}(\mathbf{x}) \in \mathbb{C}^{B_{RX} \times B_{TX}}$ and $\mathbf{C} = \text{unvec}(\mathbf{c}) \in \mathbb{C}^{M \times T}$. If the matrix multiplication involving $\mathbf{S}$ can be implemented using the FFT, e.g., Zadoff-Chu (ZC) [29] or DFT [11] training sequence, (40) and (41) can be implemented using the FFT because $\mathbf{A}_{RX}$ and $\mathbf{A}_{TX}$ are overcomplete DFT matrices. For example, each column of $\mathbf{A}_{TX} \mathbf{X}^H$ in (40) can be computed using the B_{TX}-point FFT with pruned outputs, i.e., retaining only N outputs, because we constructed $\mathbf{A}_{TX}$ as an overcomplete DFT matrix.

In particular, the matrix multiplications involving $\mathbf{A}_{TX}$, $\mathbf{S}^H$, and $\mathbf{A}_{RX}$ in (40) can be implemented with B_{TX}-point FFT with pruned outputs repeated B_{RX} times, T-point IFFT with pruned inputs repeated B_{RX} times, and B_{RX}-point FFT with pruned outputs repeated T times, respectively.[4] Using the same logic, the matrix multiplications involving $\mathbf{S}$, $\mathbf{A}_{TX}^H$, and $\mathbf{A}_{RX}^H$ in (41) can be implemented using T-point FFT with pruned outputs repeated M times, B_{TX}-point IFFT with pruned inputs repeated M times, and B_{RX}-point IFFT with pruned inputs repeated B_{TX} times, respectively. Therefore, the complexity of the FFT-based implementation of (40) and (41) is $O(B_{RX} B_{TX} \log B_{TX} + B_{RX} T \log T + T B_{RX} \log B_{RX})$ and $O(MT \log T + M B_{TX} \log B_{TX} + B_{TX} B_{RX} \log B_{RX})$, respectively.

To illustrate the efficiency of the FFT-based implementation of (40) and (41), M/N, M/B_{RX}, M/B_{TX}, and M/T are assumed to be fixed. Then, the complexity of the FFT-based implementation of $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^H \mathbf{c}$ is $O(M^2 \log M)$,

---

[4]The inputs and outputs are pruned because $\mathbf{A}_{RX}$, $\mathbf{A}_{TX}$, and $\mathbf{S}$ are rectangular, not square. The details of the pruned FFT are presented in [30–32].

whereas the complexity of directly computing $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^H \mathbf{c}$ is $O(M^4)$. Therefore, the complexity of Algorithms 2 and 3 is reduced when $h(\mathbf{x})$ and $\nabla h(\mathbf{x})$ are implemented using the FFT operations.

Remark 3 *Line 5 of Algorithms 2 and 3 is equivalent to solving*

$$\underset{\mathbf{x}_{\mathcal{I}} \in \mathbb{C}^{|\mathcal{I}|}}{\text{argmax}} \ h_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}}) = \underset{\mathbf{x}_{\mathcal{I}} \in \mathbb{C}^{|\mathcal{I}|}}{\text{argmax}} \ (f_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}}) + g_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}})) \quad (42)$$

where

$$f_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}}) = \log Pr\left[\hat{\mathbf{y}} = Q(\sqrt{\rho} \mathbf{A}_{\mathcal{I}} \mathbf{x}_{\mathcal{I}} + \mathbf{n}) \mid \mathbf{x}_{\mathcal{I}} \right], \quad (43)$$

$$g_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}}) = -\|\mathbf{x}_{\mathcal{I}}\|^2, \quad (44)$$

and $\mathbf{A}_{\mathcal{I}} \in \mathbb{C}^{MT \times |\mathcal{I}|}$ is the collection of $\mathbf{a}_i$ with $i \in \mathcal{I}$. Therefore, only $h_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}})$ and $\nabla h_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}})$ are required in Line 5 of Algorithms 2 and 3, which are low-dimensional functions defined on $\mathbb{C}^{|\mathcal{I}|}$ where $|\mathcal{I}| = O(L)$. If $h_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}})$ and $\nabla h_{\mathcal{I}}(\mathbf{x}_{\mathcal{I}})$ are computed based on the same logic in (40) and (41) but $\mathbf{A}$ replaced by $\mathbf{A}_{\mathcal{I}}$, the complexity of Algorithms 2 and 3 is reduced further because the size of the FFT is reduced in Line 5.

5 Results and discussion

In this section, we evaluate the performance of Algorithms 2 and 3 from different aspects in terms of the accuracy, achievable rate, and complexity. Throughout this section, we consider a mmWave massive MIMO system with 1 bit ADCs, whose parameters are $M = N = 64$ and $T = 80$. The rest vary from simulation to simulation, which consist of B_{RX}, B_{TX}, and L. In addition, we consider $\mathbf{S}$, whose rows are the circular shifts of the ZC training sequence of length T as in [15, 33]. Furthermore, $\mathbf{H}$ is either random or deterministic. If $\mathbf{H}$ is random, $\alpha_\ell \sim \mathcal{CN}(0, 1)$, $\theta_{RX,\ell} \sim \text{unif}([-\pi/2, \pi/2])$, and $\theta_{TX,\ell} \sim \text{unif}([-\pi/2, \pi/2])$ are independent. Otherwise, we consider different $\mathbf{H}$ from simulation to simulation.

The MSEs of $\{\alpha_\ell\}_{\ell=1}^L$, $\{\theta_{RX,\ell}\}_{\ell=1}^L$, and $\{\theta_{TX,\ell}\}_{\ell=1}^L$ are

$$\text{MSE}(\{\alpha_\ell\}_{\ell=1}^L) = \mathbb{E}\left\{ \frac{1}{L} \sum_{\ell=1}^L |\tilde{\alpha}_\ell - \alpha_\ell|^2 \right\}, \quad (45)$$

$$\text{MSE}(\{\theta_{RX,\ell}\}_{\ell=1}^L) = \mathbb{E}\left\{ \frac{1}{L} \sum_{\ell=1}^L (\tilde{\theta}_{RX,\ell} - \theta_{RX,\ell})^2 \right\}, \quad (46)$$

$$\text{MSE}(\{\theta_{TX,\ell}\}_{\ell=1}^L) = \mathbb{E}\left\{ \frac{1}{L} \sum_{\ell=1}^L (\tilde{\theta}_{TX,\ell} - \theta_{TX,\ell})^2 \right\} \quad (47)$$

where $(\tilde{\alpha}_\ell, \tilde{\theta}_{RX,\ell}, \tilde{\theta}_{TX,\ell})$ corresponds to some non-zero element of $\tilde{\mathbf{X}} = \text{unvec}(\tilde{\mathbf{x}}) \in \mathbb{C}^{B_{RX} \times B_{TX}}$. The normalized MSE (NMSE) of $\mathbf{H}$ is

$$\text{NMSE}(\mathbf{H}) = \mathbb{E}\left\{ \frac{\|\tilde{\mathbf{H}} - \mathbf{H}\|_F^2}{\|\mathbf{H}\|_F^2} \right\} \quad (48)$$

where $\tilde{\mathbf{H}} = \mathbf{A}_{\mathrm{RX}}\tilde{\mathbf{X}}\mathbf{A}_{\mathrm{TX}}$. In (45)–(48), the symbol ˜ is used to emphasize that the quantity is an estimate.

Throughout this section, we consider the debiasing variant of Algorithm 2. The halting condition of Algorithms 2 and 3 is to halt when the current and previous supp($\tilde{\mathbf{x}}$) are the same. The gradient descent method is used to solve convex optimization problems, which consist of (35) and Line 5 of Algorithms 2 and 3. The backtracking line search is used to compute the step size in the gradient descent method and κ in Line 3 of Algorithm 3. In addition, η is selected so that Conjecture 1 is satisfied. In this paper, we select the maximum η satisfying

$$\min_{i \in \{1,2,\ldots,B\}} |B_\eta(i)| > 1. \tag{49}$$

For example, the maximum η satisfying (49) is $\eta = 0.6367$ when $B_{\mathrm{RX}} = 2M$ and $B_{\mathrm{TX}} = 2N$. The channel estimation criterion of Algorithms 2 and 3 is either MAP or ML, which depends on whether $\mathbf{H}$ is random or deterministic. To compare the BMS-based and non-BMS-based

algorithms, the performance of the GraSP and GraHTP algorithms is shown as a reference in Figs. 4, 5, 6, and 7. The GraSP and GraHTP algorithms forbid $B_{\mathrm{RX}} \gg M$ and $B_{\mathrm{TX}} \gg N$ because the GraSP and GraHTP algorithms diverge when $\mathbf{A}$ is highly coherent. Therefore, the parameters are selected as $B_{\mathrm{RX}} = M$ and $B_{\mathrm{TX}} = N$ when the GraSP and GraHTP algorithms are implemented. Such B_{RX} and B_{TX}, however, are problematic because the mismatch in (7) is inversely proportional to B_{RX} and B_{TX}.

In Figs. 4 and 5, we compare the accuracy of the BMS-based and band excluding-based (BE-based) algorithms at different SNRs using MSE($\{\alpha_\ell\}_{\ell=1}^L$), MSE($\{\theta_{\mathrm{RX},\ell}\}_{\ell=1}^L$), MSE($\{\theta_{\mathrm{TX},\ell}\}_{\ell=1}^L$), and NMSE($\mathbf{H}$). The BE hard thresholding technique was proposed in [26], which was applied to the orthogonal matching pursuit (OMP) algorithm [34]. In this paper, we apply the BE hard thresholding technique to the GraSP algorithm, which results in the BEGraSP algorithm. In this example, $B_{\mathrm{RX}} = B_{\mathrm{TX}} = 256$ for the BMS-based and BE-based algorithms. We assume that
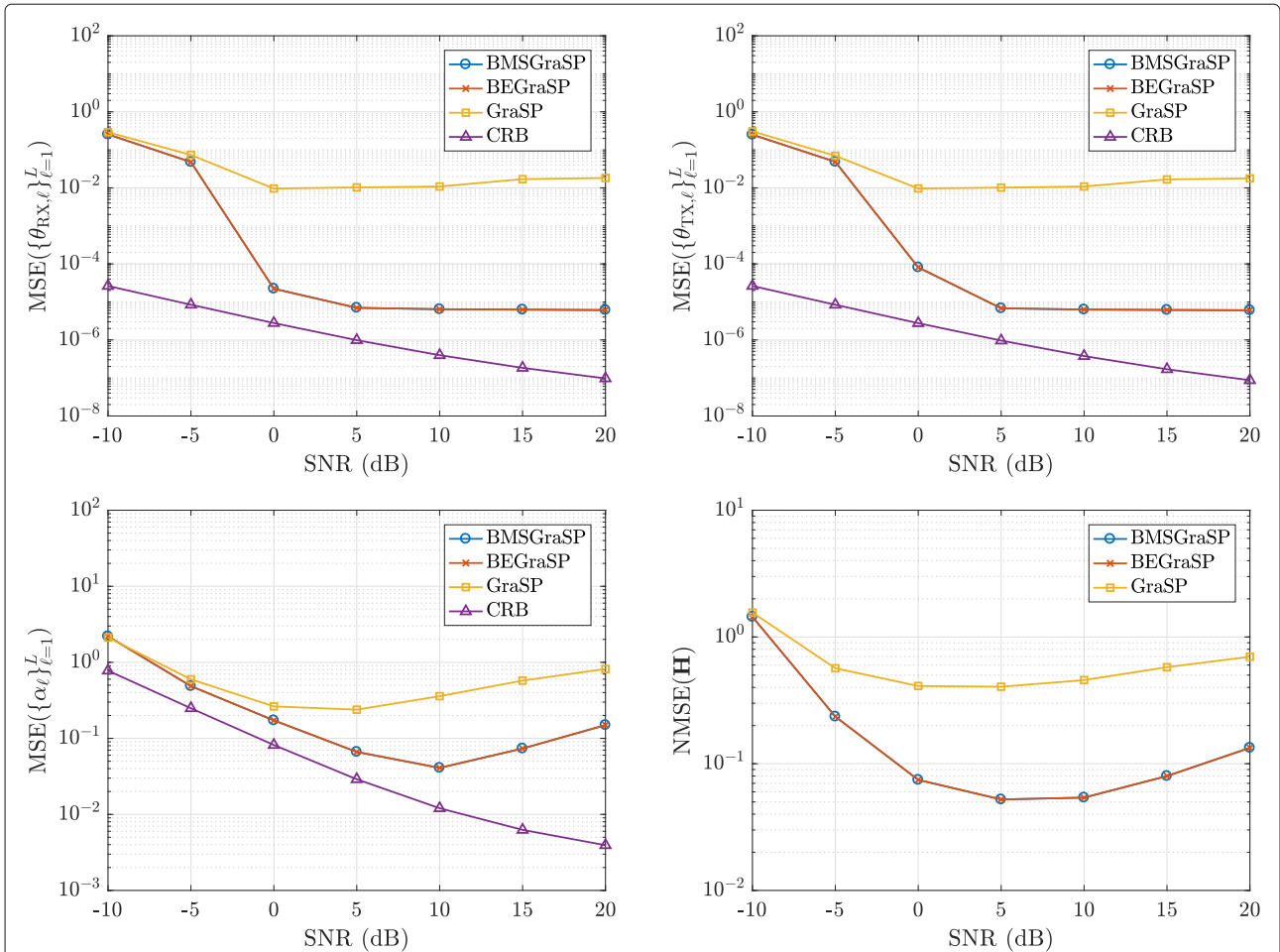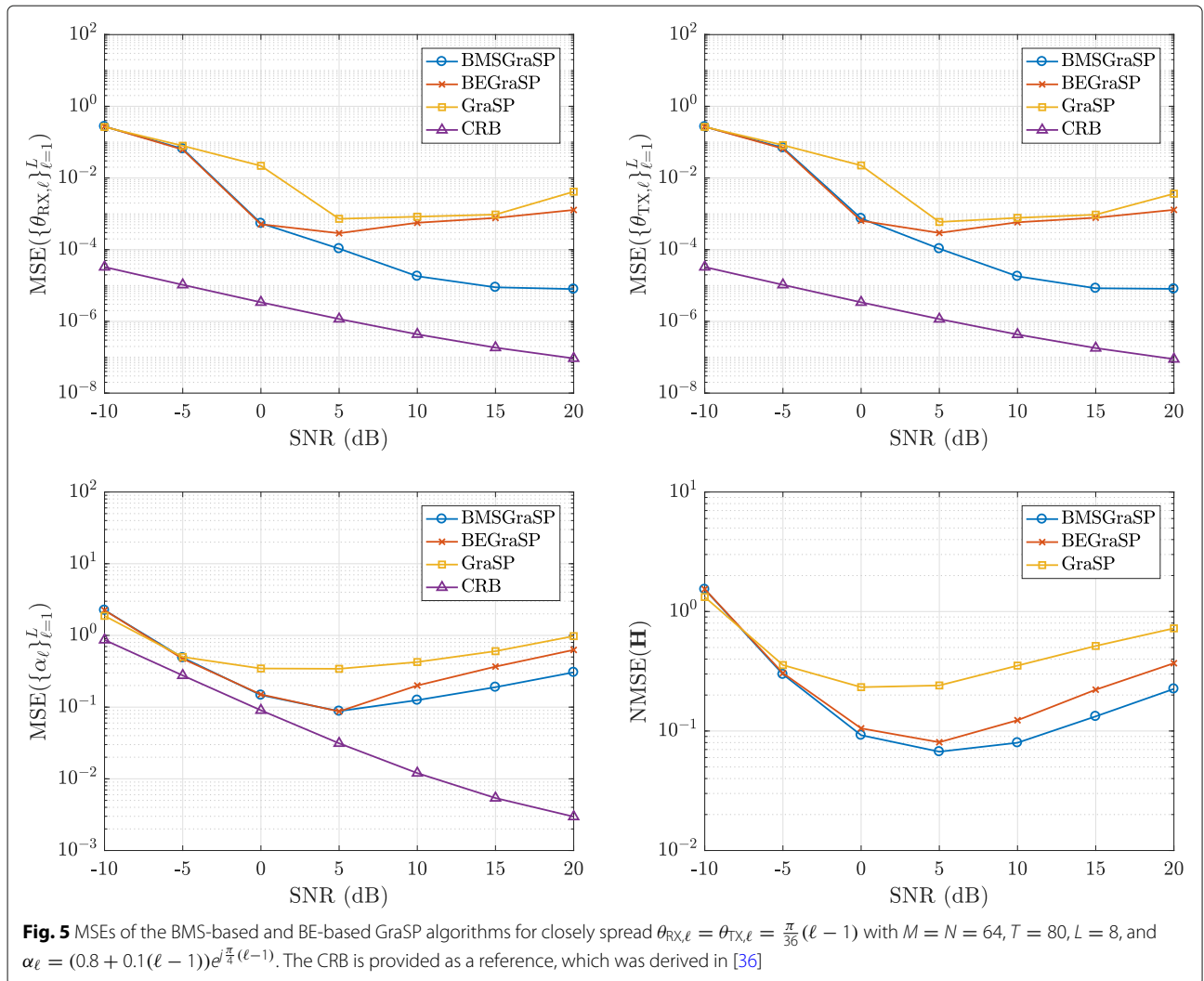

Fig. 4 MSEs of the BMS-based and BE-based GraSP algorithms for widely spread $\theta_{\mathrm{RX},\ell} = \theta_{\mathrm{TX},\ell} = \frac{\pi}{18}(\ell - 1)$ with $M = N = 64$, $T = 80$, $L = 8$, and $\alpha_\ell = (0.8 + 0.1(\ell - 1))e^{j\frac{\pi}{4}(\ell-1)}$. The CRB is provided as a reference, which was derived in [36]

Fig. 5 MSEs of the BMS-based and BE-based GraSP algorithms for closely spread $\theta_{\text{RX},\ell} = \theta_{\text{TX},\ell} = \frac{\pi}{36}(\ell - 1)$ with $M = N = 64$, $T = 80$, $L = 8$, and $\alpha_\ell = (0.8 + 0.1(\ell - 1))e^{j\frac{\pi}{4}(\ell-1)}$. The CRB is provided as a reference, which was derived in [36]

$L = 8$ and $\mathbf{H}$ is deterministic where $\alpha_\ell = (0.8 + 0.1(\ell - 1))e^{j\frac{\pi}{4}(\ell-1)}$. However, $\{\theta_{\text{RX},\ell}\}_{\ell=1}^L$ and $\{\theta_{\text{TX},\ell}\}_{\ell=1}^L$ vary from simulation to simulation, which are either widely spread (Fig. 4) or closely spread (Fig. 5). In Figs. 4 and 5, the notion of widely and closely spread paths refer to the fact that the minimum 2-norm distance between the paths are either relatively far or close, i.e., $\min_{i\neq j} \|(\theta_{\text{RX},i} - \theta_{\text{RX},j}, \theta_{\text{TX},i} - \theta_{\text{TX},j})\|_2$ of Fig. 4, which is $\sqrt{(\pi/18)^2 + (\pi/18)^2}$, is greater than that of Fig. 5, which is $\sqrt{(\pi/36)^2 + (\pi/36)^2}$. The path gains, AoAs, and AoDs are assumed to be deterministic because the CRB is defined for deterministic parameters only [35]. A variant of the CRB for random parameters is known as the Bayesian CRB, but adding the Bayesian CRB to our work is left as a future work because applying the Bayesian CRB to non-linear measurements, e.g., 1 bit ADCs, is not as straightforward.

According to Figs. 4 and 5, the BMS-based algorithms succeed to estimate both widely spread and closely spread

paths, whereas the BE-based algorithms fail to estimate closely spread paths. This follows because the BE hard thresholding technique was derived based on the assumption that $\text{supp}(\mathbf{x}^*)$ is widely spread. In contrast, the BMS hard thresholding technique is proposed based on Conjecture 1 without any assumption on $\text{supp}(\mathbf{x}^*)$. This means that when $\text{supp}(\mathbf{x}^*)$ is closely spread, the BE hard thresholding technique cannot properly estimate $\text{supp}(\mathbf{x}^*)$ because the BE hard thresholding technique, by its nature, excludes the elements near the maximum element of $\mathbf{x}^*$ from its potential candidate. The BMS hard thresholding technique, in contrast, uses the elements near the maximum element of $\mathbf{x}^*$ to construct the by-product testing set only, i.e., Line 4 of Algorithm 1. Therefore, the BMS-based algorithms are superior to the BE-based algorithms when the paths are closely spread. The Cramér-Rao bounds (CRBs) of $\text{MSE}(\{\alpha_\ell\}_{\ell=1}^L)$, $\text{MSE}(\{\theta_{\text{RX},\ell}\}_{\ell=1}^L)$, and $\text{MSE}(\{\theta_{\text{TX},\ell}\}_{\ell=1}^L)$ are provided, which were derived in [36]. The gaps between
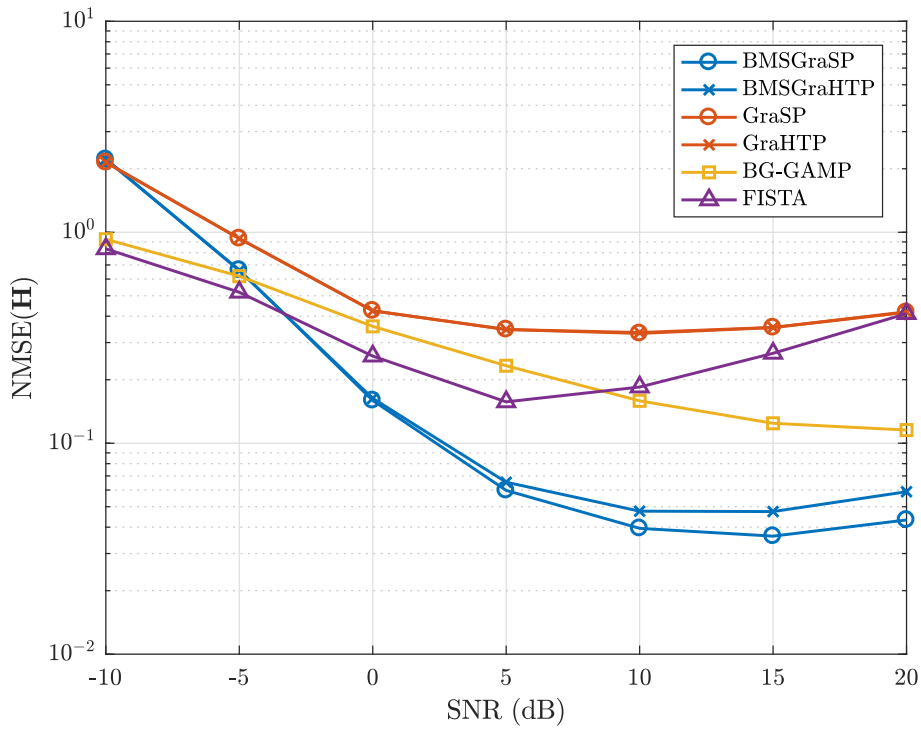
Fig. 6 NMSE vs. SNR where $M = N = 64$, $T = 80$, and $L = 4$ with varying B_{RX} and B_{TX} from algorithm to algorithm
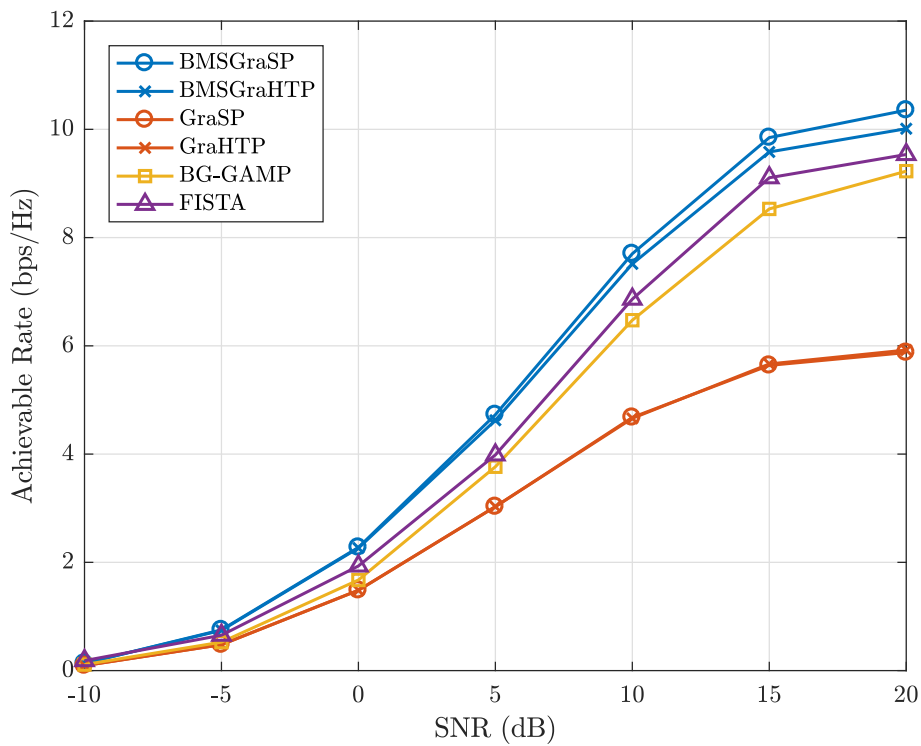

Fig. 7 Achievable rate lower bound [15] vs. SNR where $M = N = 64$, $T = 80$, and $L = 4$ with varying B_{RX} and B_{TX} from algorithm to algorithm

the MSEs and their corresponding CRBs can be interpreted as a performance limit incurred by the discretized AoAs and AoDs. To overcome such limit, the AoAs and AoDs must be estimated based on the off-grid method, which is beyond the scope of this paper.

In addition, note that $\text{MSE}(\{\alpha_\ell\}_{\ell=1}^L)$ and $\text{NMSE}(\mathbf{H})$ worsen as the SNR enters the high SNR regime. To illustrate why $\mathbf{x}^*$ cannot be estimated in the high SNR regime in 1 bit ADCs, note that

$$\begin{aligned}Q(\sqrt{\rho}\mathbf{A}\mathbf{x}^* + \mathbf{n}) &\approx Q(\sqrt{\rho}\mathbf{A}\mathbf{x}^*)\\ &= Q(\sqrt{\rho}\mathbf{A}c\mathbf{x}^*)\end{aligned} \tag{50}$$

in the high SNR regime with $c > 0$, which means that $\mathbf{x}^*$ and $c\mathbf{x}^*$ are indistinguishable because the magnitude information is lost by 1 bit ADCs. The degradation of the recovery accuracy in the high SNR regime with 1 bit ADCs is an inevitable phenomenon, as observed from other previous works on low-resolution ADCs [11, 14, 15, 33, 37].

In Figs. 6 and 7, we compare the performance of Algorithms 2 and 3, and other estimators when $\mathbf{H}$ is random. The Bernoulli Gaussian-GAMP (BG-GAMP) algorithm [15] is an iterative approximate MMSE estimator of $\mathbf{x}^*$, which was derived based on the assumption that x_i^* is distributed as $\mathcal{CN}(0, 1)$ with probability L/B but 0 otherwise, namely, the BG distribution. The fast iterative shrinkage-thresholding algorithm (FISTA) [38] is an iterative MAP estimator of $\mathbf{x}^*$, which was derived based on the assumption that the logarithm of the PDF of $\mathbf{x}^*$ is $g_{\text{FISTA}}(\mathbf{x}) = -\gamma \|\mathbf{x}\|_1$ ignoring the constant factor, namely, the Laplace distribution. Therefore, the estimate of $\mathbf{x}^*$ is

$$\underset{\mathbf{x} \in \mathbb{C}^B}{\arg\max}\, (f(\mathbf{x}) + g_{\text{FISTA}}(\mathbf{x})), \tag{51}$$

which is solved using the accelerated proximal gradient descent method [38]. The regularization parameter γ is selected so that the expected sparsity of (51) is $3L$ for a fair comparison, which was suggested in [17]. In this example, $L = 4$, whereas B_{RX} and B_{TX} vary from algorithm to algorithm. In particular, we select $B_{\text{RX}} = B_{\text{TX}} = 256$ for Algorithms 2, 3, and the FISTA, whereas $B_{\text{RX}} = M$ and $B_{\text{TX}} = N$ for the BG-GAMP algorithm.

In Fig. 6, we compare the accuracy of Algorithms 2, 3, and other estimators at different SNRs using $\text{NMSE}(\mathbf{H})$. According to Fig. 6, Algorithms 2 and 3 outperform the BG-GAMP algorithm and FISTA as the SNR enters the medium SNR regime. The accuracy of the BG-GAMP algorithm is disappointing because the mismatch in (7) is inversely proportional to B_{RX} and B_{TX}. However, increasing B_{RX} and B_{TX} is forbidden because the BG-GAMP algorithm diverges when $\mathbf{A}$ is highly coherent. The accuracy of the FISTA is disappointing because the Laplace distribution does not match the distribution of $\mathbf{x}^*$. Note that (23), which is the basis of Algorithms 2 and 3, is

indeed the MAP estimate of $\mathbf{x}^*$, which is in contrast to the FISTA. According to Fig. 6, $\text{NMSE}(\mathbf{H})$ worsens as the SNR enters the high SNR regime, which follows from the same reason as in Figs. 4 and 5.

In Fig. 7, we compare the achievable rate lower bound of Algorithms 2, 3, and other estimators at different SNRs when the precoders and combiners are selected based on $\tilde{\mathbf{H}}$. The achievable rate lower bound shown in Fig. 7 is presented in [15], which was derived based on the Bussgang decomposition [13] in conjunction with the fact that the worst-case noise is Gaussian. According to Fig. 7, Algorithms 2 and 3 outperform the BG-GAMP algorithm and FISTA, which is consistent with the result in Fig. 6.

In Fig. 8, we compare the complexity of Algorithms 2, 3, and other estimators at different B_{RX} and B_{TX} when $\mathbf{H}$ is random. To analyze the complexity, note that Algorithms 2, 3, and the FISTA require $h(\mathbf{x})$ and $\nabla h(\mathbf{x})$ at each iteration, whose bottlenecks are $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^H\mathbf{c}$, respectively, while the BG-GAMP algorithm requires $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^H\mathbf{c}$ at each iteration. Therefore, the complexity is measured based on the number of complex multiplications performed to compute $\mathbf{A}\mathbf{x}$ and $\mathbf{A}^H\mathbf{c}$, which are implemented based on the FFT. In this example, $L = 4$, whereas SNR is either 0 dB or 10 dB.

In this paper, the complexity of the BG-GAMP algorithm is used as a baseline because the BG-GAMP algorithm is widely used. The normalized complexity is defined as the number of complex multiplications performed divided by the per-iteration complexity of the BG-GAMP. For example, the normalized complexity of the FISTA with $B_{\text{RX}} = B_{\text{TX}} = 256$ is 160 when the complexity of the FISTA with $B_{\text{RX}} = B_{\text{TX}} = 256$ is equivalent to the complexity of the 160-iteration BG-GAMP algorithm with $B_{\text{RX}} = B_{\text{TX}} = 256$. In practice, the BG-GAMP algorithm converges in 15 iterations when $\mathbf{A}$ is incoherent [39]. In this paper, an algorithm is said to be as efficient as the BG-GAMP algorithm when the normalized complexity is below the target threshold, which is 15. As a sidenote, our algorithms, namely, the BMSGraSP and BMSGraHTP, requires 2.1710 and 2.0043 iterations in average, respectively, across the entire SNR range.

According to Fig. 8, the complexity of the FISTA is impractical because the objective function of (51) is a high-dimensional function defined on $\mathbb{C}^B$ where $B \gg MN$. In contrast, the complexity of Algorithms 2 and 3 is dominated by (42), whose objective function is a low-dimensional function defined on $\mathbb{C}^{|\mathcal{I}|}$ where $|\mathcal{I}| = O(L)$. The normalized complexity of Algorithms 2 and 3 is below the target threshold when $B_{\text{RX}} \geq 192$ and $B_{\text{TX}} \geq 192$. Therefore, we conclude that Algorithms 2 and 3 are as efficient as the BG-GAMP algorithm when $B_{\text{RX}} \gg M$ and $B_{\text{TX}} \gg N$.
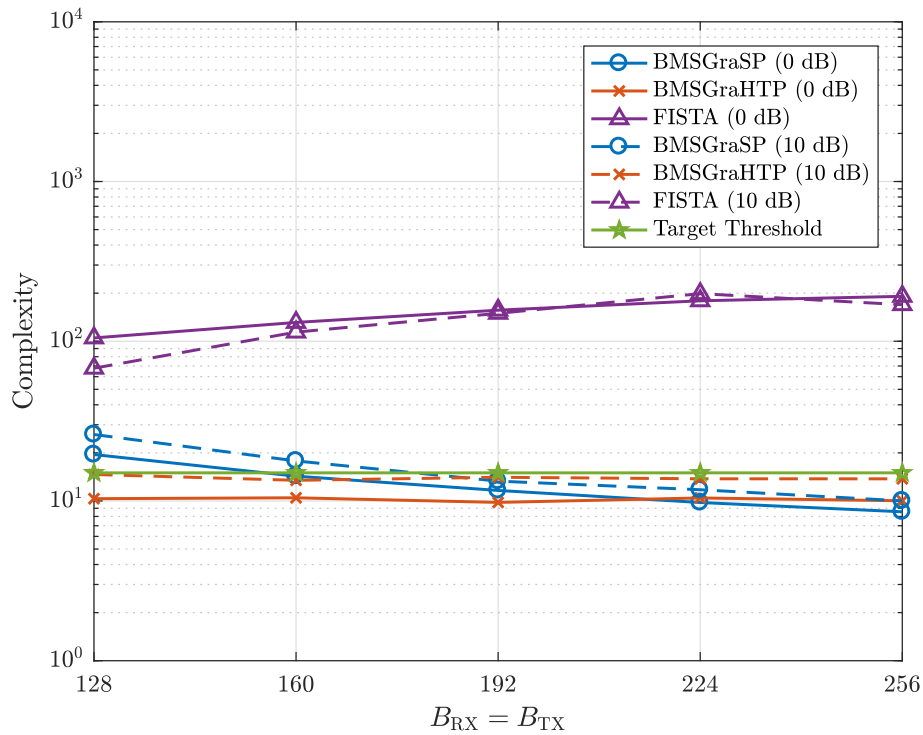
Fig. 8 Normalized complexity vs. $B_{RX} = B_{TX}$ where $M = N = 64$, $T = 80$, and $L = 4$ at SNR $= 0$ dB and SNR $= 10$ dB

6   Conclusions

In the mmWave band, the channel estimation problem is converted to a sparsity-constrained optimization problem, which is NP-hard to solve. To approximately solve sparsity-constrained optimization problems, the GraSP and GraHTP algorithms were proposed in CS, which pursue the gradient of the objective function. The GraSP and GraHTP algorithms, however, break down when the objective function is ill-conditioned, which is incurred by the highly coherent sensing matrix. To remedy such break down, we proposed the BMS hard thresholding technique, which is applied to the GraSP and GraHTP algorithms, namely, the BMSGraSP and BMSGraHTP algorithms, respectively. Instead of directly hard thresholding the gradient of the objective function, the BMS-based algorithms test whether an index is the ground truth index or the by-product of another index. We also proposed an FFT-based fast implementation of the BMS-based algorithms, whose complexity is reduced from $O(M^4)$ to $O(M^2 \log M)$. In the simulation, we compared the performance of the BMS-based, BE-based, BG-GAMP, and FISTA algorithms from different aspects in terms of the accuracy, achievable rate, and complexity. The BMS-based algorithms were shown to outperform other estimators, which proved to be both accurate and efficient. Our algorithms, however, addressed only the flat fading scenario, so an interesting future work would be to propose a low-complexity channel estimator capable of dealing with the wideband scenario.

7   Methods/experimental

The aim of this study is to propose an accurate yet efficient channel estimator for mmWave massive MIMO systems with 1 bit ADCs. Our channel estimator was proposed based on theoretical analysis. To be specific, we adopted and modified CS algorithms to exploit the sparse nature of the mmWave virtual channels. In addition, we carefully analyzed the proposed channel estimator to reduce the complexity. To verify the accuracy and complexity of the proposed channel estimator, we conducted extensive (Monte-Carlo) simulations.

Abbreviations
ADC: Analog-to-digital converter; AoA: Angle-of-arrival; AoD: Angle-of-departure; AWGN: Additive white Gaussian noise; BE: Band excluding; BG-GAMP: Bernoulli Gaussian-generalized approximate message passing; BLMMSE: Bussgang linear minimum mean squared error; BMS: Band maximum selecting; CoSaMP: Compressive sampling matching pursuit; CRB: Cramér-Rao bound; CS: Compressive sensing; DFT: Discrete fourier transform; FFT: Fast Fourier transform; FISTA: Fast iterative shrinkage-thresholding algorithm; GAMP: Generalized approximate message passing; GEC-SR: Generalized expectation consistent signal recovery; GraHTP: Gradient hard thresholding pursuit; GraSP: Gradient support pursuit; HTP: Hard thresholding pursuit; i.i.d.: Independent and identically distributed; LOS: Line-of-sight; MAP: Maximum a posteriori; MIMO: Multiple-input multiple-output; mmWave: Millimeter wave; nML: Near maximum likelihood; NMSE: Normalized mean squared error; OMP: Orthogonal matching pursuit; RIP: Restricted isometry property; SNR: Signal-to-noise ratio; ULA: Uniform linear array; ZF: Zadoff-Chu

References

1. A. L. Swindlehurst, E. Ayanoglu, P. Heydari, F. Capolino, Millimeter-wave massive MIMO: the next wireless revolution? IEEE Commun. Mag. **52**(9), 56–62 (2014). https://doi.org/10.1109/MCOM.2014.6894453

2. Z. Gao, L. Dai, D. Mi, Z. Wang, M. A. Imran, M. Z. Shakir, mmWave massive-MIMO-based wireless backhaul for the 5G Ultra-dense network. IEEE Wirel. Commun. **22**(5), 13–21 (2015). https://doi.org/10.1109/MWC.2015.7306533

3. T. E. Bogale, L. B. Le, Massive MIMO and mmWave for 5G Wireless HetNet: potential benefits and challenges. IEEE Veh. Technol. Mag. **11**(1), 64–75 (2016). https://doi.org/10.1109/MVT.2015.2496240

4. F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, P. Popovski, Five disruptive technology directions for 5G. IEEE Commun. Mag. **52**(2), 74–80 (2014). https://doi.org/10.1109/MCOM.2014.6736746

5. Bin Le, T. W. Rondeau, J. H. Reed, C. W. Bostian, Analog-to-digital converters. IEEE Sig. Proc. Mag. **22**(6), 69–77 (2005). https://doi.org/10.1109/MSP.2005.1550190

6. C. Mollén, J. Choi, E. G. Larsson, R. W. Heath, Uplink performance of wideband massive MIMO with one-bit ADCs. IEEE Trans. Wirel. Commun. **16**(1), 87–100 (2017). https://doi.org/10.1109/TWC.2016.2619343

7. L. Fan, S. Jin, C. Wen, H. Zhang, Uplink achievable rate for massive MIMO systems with low-resolution ADC. IEEE Commun. Lett. **19**(12), 2186–2189 (2015). https://doi.org/10.1109/LCOMM.2015.2494600

8. J. Zhang, L. Dai, S. Sun, Z. Wang, On the spectral efficiency of massive MIMO systems with low-resolution ADCs. IEEE Commun. Lett. **20**(5), 842–845 (2016). https://doi.org/10.1109/LCOMM.2016.2535132

9. S. Jacobsson, G. Durisi, M. Coldrey, U. Gustavsson, C. Studer, Throughput analysis of massive MIMO uplink with low-resolution ADCs. IEEE Trans. Wirel. Commun. **16**(6), 4038–4051 (2017). https://doi.org/10.1109/TWC.2017.2691318

10. J. Choi, J. Mo, R. W. Heath, Near maximum-likelihood detector and channel estimator for uplink multiuser massive MIMO systems with one-bit ADCs. IEEE Trans. Commun. **64**(5), 2005–2018 (2016). https://doi.org/10.1109/TCOMM.2016.2545666

11. Y. Li, C. Tao, G. Seco-Granados, A. Mezghani, A. L. Swindlehurst, L. Liu, Channel estimation and performance analysis of one-bit massive MIMO systems. IEEE Trans. Sign. Proc. **65**(15), 4075–4089 (2017). https://doi.org/10.1109/TSP.2017.2706179

12. D. P. Bertsekas, Nonlinear Programming. Journal of the Operational Research Society. **48**(3), 334–334 (1997). https://doi.org/10.1057/palgrave.jors.2600425

13. J. J. Bussgang, Research Laboratory of Electronics, Massachusetts Institute of Technology. Technical report. **216**, 1–14 (1952)
    Article type: Technical report Institution: Research Laboratory of Electronics, Massachusetts Institute of Technology Volume: 216 Page: 1-14 Year: 1952

14. H. He, C. Wen, S. Jin, Bayesian optimal data detector for hybrid mmWave MIMO-OFDM systems with low-resolution ADCs. IEEE J. Sel. Top. Sig. Proc. **12**(3), 469–483 (2018). https://doi.org/10.1109/JSTSP.2018.2818063

15. J. Mo, P. Schniter, R. W. Heath, Channel estimation in broadband millimeter wave MIMO systems with Few-Bit ADCs. IEEE Trans. Sign. Proc. **66**(5), 1141–1154 (2018). https://doi.org/10.1109/TSP.2017.2781644

16. T. Liu, C. Wen, S. Jin, X. You, in *2016 IEEE International Symposium on Information Theory (ISIT)*. Generalized turbo signal recovery for nonlinear measurements and orthogonal sensing matrices, (2016), pp. 2883–2887. https://doi.org/10.1109/ISIT.2016.7541826

17. S. Bahmani, B. Raj, P. T. Boufounos, Greedy Sparsity-Constrained Optimization. J. Mach. Learn. Res. **14**(Mar), 807–841 (2013)

18. X.-T. Yuan, P. Li, T. Zhang, Gradient Hard Thresholding Pursuit. J. Mach. Learn. Res. **18**(166), 1–43 (2018)

19. D. Needell, J. A. Tropp, CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. Appl. Comput. Harmon. Anal. **26**(3), 301–321 (2009). https://doi.org/10.1016/j.acha.2008.07.002

20. S. Foucart, Hard Thresholding Pursuit: An Algorithm for Compressive Sensing. SIAM J. Numer. Anal. **49**(6), 2543–2563 (2011). https://doi.org/10.1137/100806278

21. M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, E. Erkip, Millimeter wave channel modeling and cellular capacity evaluation. IEEE J. Sel. Areas Commun. **32**(6), 1164–1179 (2014). https://doi.org/10.1109/JSAC.2014.2328154

22. A. M. Sayeed, Deconstructing Multiantenna Fading Channels. IEEE Trans. Sign. Proc. **50**(10), 2563–2579 (2002). https://doi.org/10.1109/TSP.2002.803324

23. W. Hong, K. Baek, Y. Lee, Y. Kim, S. Ko, Study and Prototyping of Practically Large-Scale mmWave Antenna Systems for 5G Cellular Devices. IEEE Commun. Mag. **52**(9), 63–69 (2014). https://doi.org/10.1109/MCOM.2014.6894454

24. S. Boyd, L. Vandenberghe, *Convex Optimization*. (Cambridge University Press, Cambridge, 2004)

25. Y. C. Eldar, G. Kutyniok, *Compressed Sensing: Theory and Applications*. (Cambridge University Press, Cambridge, 2012)

26. A. Fannjiang, W. Liao, Coherence Pattern–Guided Compressive Sensing with Unresolved Grids. SIAM J. Imaging Sci. **5**(1), 179–202 (2012). https://doi.org/10.1137/110838509

27. N. Jindal, MIMO broadcast channels with finite-rate feedback. IEEE Trans. Inf. Theory. **52**(11), 5045–5060 (2006). https://doi.org/10.1109/TIT.2006.883550

28. Z. Marzi, D. Ramasamy, U. Madhow, Compressive channel estimation and tracking for large arrays in mm-wave Picocells. IEEE J. Sel. Top. Sig. Proc. **10**(3), 514–527 (2016). https://doi.org/10.1109/JSTSP.2016.2520899

29. D. Chu, Polyphase Codes with Good Periodic Correlation Properties (Corresp.) IEEE Trans. Inf. Theory. **18**(4), 531–532 (1972). https://doi.org/10.1109/TIT.1972.1054840

30. J. Markel, FFT Pruning. IEEE Trans. Audio Electroacoustics. **19**(4), 305–311 (1971). https://doi.org/10.1109/TAU.1971.1162205

31. D. Skinner, Pruning the Decimation In-Time FFT Algorithm. IEEE Trans. Acoust. Speech Sig. Proc. **24**(2), 193–194 (1976). https://doi.org/10.1109/TASSP.1976.1162782

32. T. Sreenivas, P. Rao, FFT Algorithm for both input and output pruning. IEEE Trans. Acoust. Speech Sig. Proc. **27**(3), 291–292 (1979). https://doi.org/10.1109/TASSP.1979.1163246

33. Y. Ding, S. Chiu, B. D. Rao, Bayesian Channel estimation algorithms for massive MIMO systems with hybrid analog-digital processing and low-resolution ADCs. IEEE J. Sel. Top. Sig. Proc. **12**(3), 499–513 (2018). https://doi.org/10.1109/JSTSP.2018.2814008

34. J. A. Tropp, A. C. Gilbert, Signal Recovery from Random Measurements via Orthogonal Matching Pursuit. IEEE Trans. Inf. Theory. **53**(12), 4655–4666 (2007). https://doi.org/10.1109/TIT.2007.909108

35. H. V. Poor, *An Introduction to Signal Detection and Estimation*. (Springer, Berlin, 2013)

36. P. Wang, J. Li, M. Pajovic, P. T. Boufounos, P. V. Orlik, in *2017 51st Asilomar Conference on Signals, Systems, and Computers*. On Angular-Domain Channel Estimation for One-Bit Massive MIMO Systems with Fixed and Time-Varying Thresholds, (2017), pp. 1056–1060. https://doi.org/10.1109/ACSSC.2017.8335511

37. R. P. David, J. Cal-Braz, in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Feedback-Controlled Channel Estimation with Low-Resolution ADCs in Multiuser MIMO Systems, (2019), pp. 4674–4678. https://doi.org/10.1109/ICASSP.2019.8683652

38. A. Beck, M. Teboulle, A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. SIAM J. Imaging Sci. **2**(1), 183–202 (2009). https://doi.org/10.1137/080716542

39. J. P. Vila, P. Schniter, Expectation-maximization Gaussian-mixture approximate message passing. IEEE Trans. Sign. Proc. **61**(19), 4658–4672 (2013). https://doi.org/10.1109/TSP.2013.2272287

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.