



Article

# On Matrix Completion-Based Channel Estimators for Massive MIMO Systems

Mingjun Ding <sup>1,2</sup> , Xiaodong Yang <sup>1,2,\*</sup>, Rui Hu <sup>3,\*</sup> , Zhitao Xiao <sup>1,2</sup>, Jun Tong <sup>3</sup> and Jiangtao Xi <sup>3</sup>

<sup>1</sup> Tianjin Key Laboratory of Optoelectronic Detection Technology and System, Tianjin Polytechnic University, Tianjin 300387, China; dingmingjun@tjpu.edu.cn (M.D.); xiaozhitao@tjpu.edu.cn (Z.X.)

<sup>2</sup> School of Electronics and Information Engineering, Tianjin Polytechnic University, Tianjin 300387, China

<sup>3</sup> School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, Wollongong, NSW 2522, Australia; jtong@uow.edu.au (J.T.); jiangtao@uow.edu.au (J.X.)

\* Correspondence: 1730091236@stu.tjpu.edu.cn (X.Y.); rh546@uowmail.edu.au (R.H.)

Received: 19 October 2019; Accepted: 4 November 2019; Published: 6 November 2019



**Abstract:** Large-scale symmetric arrays such as uniform linear arrays (ULA) have been widely used in wireless communications for improving spectrum efficiency and reliability. Channel state information (CSI) is critical for optimizing massive multiple-input multiple-output (MIMO)-based wireless communication systems. The acquisition of CSI for massive MIMO faces challenges such as training shortage and high computational complexity. For millimeter wave MIMO systems, the low-rankness of the channel can be utilized to address the challenge of training shortage. In this paper, we compared several channel estimation schemes based on matrix completion (MC) for symmetrical arrays. Performance and computational complexity are discussed and compared. By comparing the performance in different scenarios, we concluded that the generalized conditional gradient with alternating minimization (GCG-Alt) estimator provided a low-cost, robust solution, while the alternating direction method of multipliers (ADMM)-based hybrid methods achieved the best performance when the array response was perfectly known.

**Keywords:** low-rankness; massive MIMO; matrix completion; compressive sensing

## 1. Introduction

Millimeter-wave (mmWave) wireless communications have drawn great attention in the industry and academia [1] thanks to the large bandwidth available in the 30–300 GHz band. To compensate for the significant path loss in this band and also thanks to the short wavelength, massive MIMO have been suggested for mmWave systems. In particular, large-scale symmetric antenna arrays, such as the uniform linear arrays (ULAs) and uniform planar arrays (UPAs) have been extensively considered for transmitters and receivers due to their neat structures and high gains for directional transmissions [2–4]. However, the coherence time in the millimeter-wave system is suggested to be short and as the number of antennas increases, the complexity of channel estimation increases. Therefore, it is challenging to acquire instantaneous channel state information (CSI) for a mmWave massive MIMO.

MmWave channels are often dominated by a small number of propagation paths, indicating that the channel is sparse in the angular domain [5]. The channel matrix can be expressed in terms of dictionary matrices, which are formed by the transmitting and receiving array response vectors, and path gains. Compressive sensing (CS) [6] can then be applied to search for the dominant paths [7–13]. Different measurement matrices can be used by choosing the precoders and combiners, as well as various recovery algorithms such as orthogonal matching pursuit (OMP) [7,14] and the adaptive CS [8–10] can be applied. In general, the above mentioned CS schemes require knowledge of the array response, which depends on the array geometry and calibration of the antenna arrays. Such

knowledge can be inaccurate when there are unknown hardware impairments, e.g., due to phase and gain errors, and imperfect calibration of the antenna arrays.

In the meantime, a small number of propagation paths also indicate that the channel is low-rank [15] and can be depicted in low-dimensional subspace. Furthermore, such low-rankness is independent of the array response and calibration errors. In [16,17], both the sparsity and low-rank property of the mmWave channel are exploited to enhance CS-based channel estimators. In [16], a two-stage estimator is proposed, where the low-rankness is exploited at the first stage while sparsity in the angular domain is exploited at the second stage. In [17], the improved alternating direction method of multipliers (ADMM) method [18] is applied to exploit the low-rankness and sparsity while at the same time enhancing the performance. These estimators, however, still require knowledge of the array response vectors. They can still suffer from performance loss when there are uncertainties in the array response.

To achieve robust channel estimation, matrix completion (MC) methods exploiting only the low-rank property of the mmWave channel have recently been proposed [19,20]. The analysis in [21] shows that the rank of the channel matrix is generally very low, which is usually much smaller than the antenna dimension. In [19], the singular value projection (SVP) algorithm [22] is adopted to solve the mmWave channel estimation problem. Later, the GCG-Alt method is developed in [20] by applying the generalized conditional gradient (GCG) framework [23] together with the alternating minimum (AltMin) method [24]. There are also other widely-studied MC algorithms that can be used for mmWave channel estimation, such as the singular value thresholding algorithm (SVT) [25] and the fixed point continuation algorithm (FPC) [26]. In this paper, we discuss several mmWave channel estimators based on MC, focusing on their performance and complexity comparisons with alternative methods based on CS. We aim to examine the pros and cons for several MC estimators and the factors that influence their performances.

The rest of this paper is organized as follows. Section 2 introduces the mmWave MIMO channel model and formulates the channel estimation problem. Section 3 introduces channel estimators based on MC, including their detailed implementation. Section 4 presents simulation results, in terms of the mean squared error (MSE) and computational complexity. Section 5 concludes the paper.

*Notation:*  $\mathbf{A}^T$ ,  $\mathbf{A}^*$ , and  $\mathbf{A}^H$  denote transpose, conjugate, and conjugate transpose of matrix  $\mathbf{A}$ , respectively.  $\|\mathbf{A}\|_1$  denotes the  $l_1$  norm.  $\mathbf{I}$  and  $\mathbf{0}$  represent the identity matrix and zero matrix/vector, respectively.  $\mathbf{A} \otimes \mathbf{B}$  and  $\mathbf{A} \odot \mathbf{B}$  denote the Kronecker product and the Hadamard product, respectively.  $\text{tr}\{\mathbf{A}\}$  is the trace of  $\mathbf{A}$  and  $\langle \mathbf{A}, \mathbf{C} \rangle = \text{tr}\{\mathbf{A}^H \mathbf{C}\}$  denotes the inner product of matrices  $\mathbf{A}$  and  $\mathbf{C}$ .  $E[\cdot]$  denotes the statistical expectation and  $\text{abs}(\cdot)$  represents taking element-wise absolute value.  $\nabla$  denotes the gradient of a function. For a matrix  $\mathbf{A} \in \mathbb{C}^{M \times N}$ ,  $\text{vec}(\mathbf{A}) \in \mathbb{C}^{MN \times 1}$  denotes the vectorization of  $\mathbf{A}$  and  $\text{vec}^{-1}(\mathbf{A}) \in \mathbb{C}^{M \times N}$  denotes the inverse of vectorization.  $\mathcal{R}(\cdot)$  and  $\mathcal{I}(\cdot)$  denote the real and imaginary part of a number or vector, respectively.  $\mathcal{CN}(a, b^2)$  denotes complex Gaussian distribution with mean  $a$  and variance  $b^2$ .

## 2. System Model

Consider a point-to-point, switch-based mmWave hybrid MIMO system, with the receiver at the mobile station (MS) shown in Figure 1. For simplicity and clarity, this paper assumes switch-based mmWave systems to investigate MC-based channel estimators. MC-based estimators can also be applied to phase shifter-based mmWave systems, when the hybrid precoders/combiners are properly designed, as shown in [20]. Therefore, the discussion in this paper can be easily extended to phase shifter-based mmWave systems. At the receiver, each of the  $N_{\text{MS}}$  antennas is equipped with a switch used to select one of the  $N_{\text{RF}}$  RF chains. The base station (BS) has the same structure with  $N_{\text{BS}}$  antennas and  $N_{\text{RF}}$  RF chains. Assume that  $N_s$  data streams are transmitted, with  $N_s \leq N_{\text{RF}} \leq \min(N_{\text{BS}}, N_{\text{MS}})$  [14,27]. The switching operation can be represented as a precoder  $\mathbf{F}$ , where the nonzero entries indicate the entries of the channel matrix that are sampled. A symbol  $\mathbf{s} \in \mathbb{C}^{N_s \times 1}$  with  $\mathbb{E}[\mathbf{s}\mathbf{s}^H] = \frac{1}{N_s} \mathbf{I}$  is precoded, resulting in the transmitted signal  $\mathbf{x} = \mathbf{F}\mathbf{s}$ . We consider a narrow-band flat fading

channel whose channel matrix  $\mathbf{H} \in \mathbb{C}^{N_{MS} \times N_{BS}}$  satisfies  $\mathbb{E} [\|\mathbf{H}\|_F^2] = N_{MS}N_{BS}$ . The received signal is expressed as:

$$\mathbf{r} = \sqrt{\rho}\mathbf{H}\mathbf{F}\mathbf{s} + \hat{\mathbf{n}}, \quad (1)$$

where  $\rho$  indicates the average received power and  $\hat{\mathbf{n}} \in \mathbb{C}^{N_{MS} \times 1}$  is a noise vector with i.i.d. entries distributed as  $\mathcal{CN}(0, \sigma_n^2)$ . Applying a combiner  $\mathbf{W}$  to the received signal at the MS, the processed received signal is given by:

$$\mathbf{y} = \sqrt{\rho}\mathbf{W}^H\mathbf{H}\mathbf{F}\mathbf{s} + \mathbf{W}^H\hat{\mathbf{n}}. \quad (2)$$

In the switch-based system, the combiner  $\mathbf{W}$  has a similar structure with the precoder  $\mathbf{F}$ .

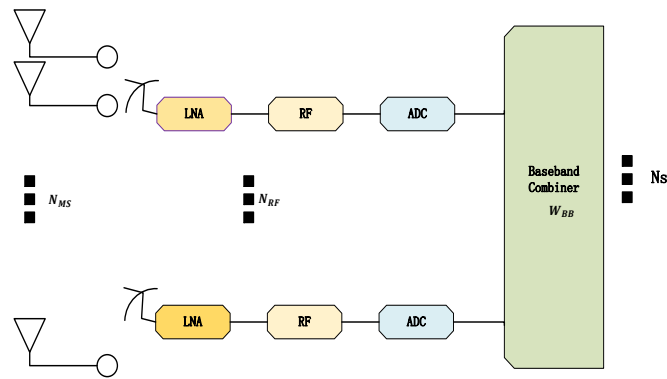


Figure 1. Switch-based mmWave (millimeter-wave) receiver.

Following [8], the ray-clustering model of  $\mathbf{H}$  is given as:

$$\mathbf{H} = \frac{1}{\sqrt{R}} \sum_{c=1}^C \sum_{r=1}^R g_{cr} \mathbf{a}_{MS}(\phi_{cr}^{MS}) \mathbf{a}_{BS}^H(\phi_{cr}^{BS}), \quad (3)$$

where  $C \sim \max\{\text{Poisson}(\lambda), 1\}$  is the number of clusters with  $\lambda$  as the mean of the Poisson distribution, and  $R$  is the number of rays in each cluster. The complex small-scale fading gain on the  $r$ -th ray of the  $c$ -th cluster is  $g_{cr}$  with  $g_{cr} \sim \mathcal{CN}(0, \gamma_c)$ , where  $\gamma_c$  is the sub-power on the  $c$ -th cluster. In Equation (3),  $\mathbf{a}_{MS}(\phi_{cr}^{MS})$  and  $\mathbf{a}_{BS}(\phi_{cr}^{BS})$  represent the array response vector for the receiver and transmitter, respectively, where  $\phi_{cr}^{MS}$  and  $\phi_{cr}^{BS}$  represent the corresponding azimuth AoA and AoD, which follow the Laplacian distribution [28].

Considering a uniform linear array (ULA) with distance between adjacent antennas being  $d$ , the array response is given by:

$$\mathbf{a}_{BS}(\phi_{cr}^{BS}) = \frac{1}{\sqrt{N_{BS}}} \left[ 1, e^{j\frac{2\pi}{\lambda_c} d \sin(\phi_{cr}^{BS})}, \dots, e^{j(N-1)\frac{2\pi}{\lambda_c} d \sin(\phi_{cr}^{BS})} \right]^T \quad (4)$$

where  $\lambda_c$  is the wavelength of the carrier wave. The array response  $\mathbf{a}_{MS}(\phi_{cr}^{MS})$  is constructed in the same manner as  $\mathbf{a}_{BS}(\phi_{cr}^{BS})$ .

### 3. Compressive Sensing-Based Channel Estimation

It has been shown that, without considering quantization errors, the mmWave channel estimation problem can be formulated as a sparse recovery problem by modeling the channel as [29–31]:

$$\mathbf{H} = \mathbf{A}_{MS} \mathbf{H}_v \mathbf{A}_{BS}^H, \quad (5)$$

where  $\mathbf{A}_{\text{MS}} = [\mathbf{a}_{\text{MS}}(\phi_1^{\text{MS}}), \dots, \mathbf{a}_{\text{MS}}(\phi_{N_1}^{\text{MS}})]$  is a unitary dictionary matrix when  $N_1 = N_{\text{MS}}$  and it is an overcomplete dictionary matrix when  $N_1 > N_{\text{MS}}$ , and  $\mathbf{A}_{\text{BS}} = [\mathbf{a}_{\text{BS}}(\phi_1^{\text{BS}}), \dots, \mathbf{a}_{\text{BS}}(\phi_{N_2}^{\text{BS}})]$  with  $N_2 \geq N_{\text{BS}}$ . Each column in  $\mathbf{A}_{\text{BS}}$  and  $\mathbf{A}_{\text{MS}}$  consists of a predefined array response vector.  $\mathbf{H}_v \in \mathbb{C}^{N_1 \times N_2}$  is a sparse matrix with only  $L$  non-zero values, with each of its non-zero values corresponding to the complex gain of a channel path. Vectorization of the channel matrix (5) produces:

$$\text{vec}(\mathbf{H}) = (\mathbf{A}_{\text{BS}}^* \otimes \mathbf{A}_{\text{MS}}) \mathbf{x}, \quad (6)$$

where  $\mathbf{x} = \text{vec}(\mathbf{H}_v)$  is a  $N_1 N_2 \times 1$  sparse vector with  $L$  non-zero values. We define  $\mathbf{\Psi} = (\mathbf{A}_{\text{BS}}^* \otimes \mathbf{A}_{\text{MS}})$  as a  $N_{\text{BS}} N_{\text{MS}} \times N_1 N_2$  dictionary matrix. Sparse recovery schemes can then be used to estimate the channel, which transforms the task of estimating  $\mathbf{H}$  to estimating the non-zero coefficients in  $\mathbf{x}$ .

A widely used method to estimate  $\mathbf{x}$  from the received signal is orthogonal matching pursuit (OMP) [7,14]. Using OMP,  $L$  path directions from the  $N_1 N_2$  candidates in the dictionary are determined. The computational complexity of the OMP method is approximately  $O(NLN_1 N_2)$ , where  $N$  is the length of the received signal. In general, a larger dictionary leads to better performance but also higher computational complexity.

The above mentioned CS approach uses a discretized approximation of the channel. It may suffer from the off-grid issue if the physical propagation paths are off the assumed grid of the angles. In this case, the number of non-zero entries in the beamspace channel  $\mathbf{H}_v$  may not be exactly equal to  $L$ , leading to a power leakage. Another challenge is that the knowledge of the array response is required, which may be imperfect in practice due to unknown hardware impairments and imperfect calibrations.

#### 4. Matrix Completion-Based Channel Estimation

In this section, we introduce MC-based estimation methods for the mmWave channel by exploiting the low-rankness of the channel matrix. By appropriately choosing the training scheme with proper precoders and combiners, the received signal provides noisy observations of a subset of the entries of  $\mathbf{H}$ :

$$[\tilde{\mathbf{Y}}]_{i,j} = \begin{cases} [\tilde{\mathbf{H}}]_{i,j}, & (i,j) \in \Omega \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where  $\tilde{\mathbf{H}} = \mathbf{H} + \mathbf{N}$  is the perturbed channel matrix,  $\mathbf{N}$  is a noise matrix,  $\Omega$  denotes a sample domain, and  $[\tilde{\mathbf{H}}]_{i,j}$  is the  $(i,j)$ th entry of  $\tilde{\mathbf{H}}$ . Define  $p = N/(N_{\text{BS}} N_{\text{MS}})$  as the sampling density, where  $N$  is the total number of samples observed.

It is discussed in [19] that when the mmWave channel matrix has strong non-coherent characteristics, it can be recovered from a subset samples of the channel matrix. We can thus formulate the channel estimation problem as a low-rank matrix completion problem as:

$$\min_{\hat{\mathbf{H}}} \text{rank}(\hat{\mathbf{H}}), \quad \text{s.t.} \quad \|P_{\Omega}(\hat{\mathbf{H}}) - P_{\Omega}(\tilde{\mathbf{H}})\|_F^2 \leq \delta_n^2, \quad (8)$$

where  $\delta_n^2$  is the error tolerance parameter and the sampling operator  $P_{\Omega}(\cdot)$  is defined as:

$$[P_{\Omega}(\tilde{\mathbf{H}})]_{i,j} = \begin{cases} [\tilde{\mathbf{H}}]_{i,j} & (i,j) \in \Omega \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

where  $[\tilde{\mathbf{H}}]_{i,j}$  denotes the  $(i,j)$ -th entry of  $\tilde{\mathbf{H}}$ . The sampling operator  $P_{\Omega}$  significantly influences the performance of the algorithm [32]. Bernoulli and uniform sampling models are proposed in [32] and a uniform spatial sampling model (USS), which improves the performance, is proposed in [33]. With USS,  $N/N_{\text{BS}}$  samples are taken for each column of the target matrix.

#### 4.1. MC Estimators

In the following, we discuss several MC estimators that can be used to solve the problem in Equation (8).

##### 4.1.1. SVT Estimator

Before presenting the SVT algorithm, let us define the matrix shrinkage operator:

$$\mathcal{S}_\tau(\mathbf{X}) = \mathbf{U}_X \mathcal{S}_\tau(\boldsymbol{\Sigma}_X) \mathbf{V}_X^H, \quad (10)$$

where  $\boldsymbol{\Sigma}_X$  denotes the singular value matrix of  $\mathbf{X}$  and  $\mathcal{S}_\tau(\boldsymbol{\Sigma}_X)$  is the element-wise shrinkage operator:

$$\mathcal{S}_\tau(x) = \begin{cases} 0, & x \leq \tau \\ x - \tau, & x > \tau \end{cases} \quad (11)$$

where  $\tau$  is the threshold.

The SVT algorithm [25] can be applied to provide a heuristic solution to Equation (8), which consists of two major steps,

$$\begin{cases} \hat{\mathbf{H}}^k = \mathcal{S}_\tau(\mathbf{X}^{k-1}) \\ \mathbf{X}^k = \mathbf{X}^{k-1} + \delta P_\Omega(\tilde{\mathbf{H}} - \hat{\mathbf{H}}^k) \end{cases} \quad (12)$$

where  $\tau > 0$ ,  $\delta$  is a step size, and  $k = 1, 2, \dots$ . The iteration is stopped when a stopping criterion is met or a maximum number of iterations  $J_{\text{SVT}}$  is reached. Singular value decomposition (SVD) is required at each iteration. Some comments regarding the implementation of the SVT algorithm to the mmWave channel estimation problem are as follows:

- The threshold is set as  $\tau = 5\sqrt{N_{\text{MS}}N_{\text{BS}}}$  following [25];
- The stepsize is set as  $\delta = 1.2/p$ ;
- Assuming the initialization  $\mathbf{X}^0 = \mathbf{0}$ ,  $\hat{\mathbf{H}}^k = \mathbf{0}$  for a small  $k < k_0$ . As such,  $\mathbf{X}^k = k\delta P_\Omega(\tilde{\mathbf{H}})$ ,  $k = 1, \dots, k_0$ . The algorithm can begin with computing  $\hat{\mathbf{H}}^{k_0}$  to save work. From [25], the integer  $k_0$  is determined by

$$\frac{\tau}{\delta \|P_\Omega(\tilde{\mathbf{H}})\|_2} \in (k_0 - 1, k_0].$$

SVT Estimator is shown in below Algorithm 1:

---

#### Algorithm 1 SVT Estimator

---

**Require:**  $P_\Omega(\tilde{\mathbf{H}})$ ,  $\delta$ ,  $\epsilon$ ,  $\tau$ ,  $J_{\text{SVT}}$ ,  $k_0$ .

- 1: Set  $\mathbf{X}^0 = k_0\delta P_\Omega(\tilde{\mathbf{H}})$ ;
  - 2: **for**  $k = 1$  to  $J_{\text{SVT}}$  **do**
  - 3:   Set  $\hat{\mathbf{H}}^k = \mathcal{S}_\tau(\mathbf{X}^{k-1})$
  - 4:   Set  $\mathbf{X}^k = \mathbf{X}^{k-1} + \delta P_\Omega(\tilde{\mathbf{H}} - \hat{\mathbf{H}}^k)$
  - 5:   **if**  $\frac{\|P_\Omega(\hat{\mathbf{H}}^k - \tilde{\mathbf{H}})\|_F}{\|P_\Omega(\tilde{\mathbf{H}})\|_F} \leq \epsilon$  **then break;**
  - 6:   **end if**
  - 7: **end for**
  - 8: **return**  $\hat{\mathbf{H}} = \hat{\mathbf{H}}^k$
- 

From [25], SVT is effective for completing large matrices with low ranks. Its performance degrades as the rank increases. The computational complexity of the SVT algorithm mainly arises from Step 3.

#### 4.1.2. FPC Estimator

The FPC algorithm [26] reformulates the MC problem using the nuclear norm, which is the summation of the singular values,

$$\min_{\hat{\mathbf{H}}} \mu \|\hat{\mathbf{H}}\|_* + \frac{1}{2} \|P_{\Omega}(\tilde{\mathbf{H}}) - P_{\Omega}(\hat{\mathbf{H}})\|_{\mathbb{F}}^2, \quad (13)$$

where  $\mu > 0$  is the regularization parameter. The algorithm consists of two steps similar to SVT:

$$\begin{cases} \mathbf{Y}^k = \hat{\mathbf{H}}^k - \delta(P_{\Omega}(\tilde{\mathbf{H}}) - P_{\Omega}(\hat{\mathbf{H}}^k)) \\ \hat{\mathbf{H}}^{k+1} = \mathcal{S}_{\delta\mu_m}(\mathbf{Y}^k) \end{cases} \quad (14)$$

where the threshold of the singular value thresholding operator is set as a variable  $\delta\mu_m$  rather than a fixed value as that in SVT. A continuous strategy [34] is used to accelerate the convergence by adapting  $\mu_m$ . The details are presented in Algorithm 2. Some comments are as below:

- We can set the step size  $\delta \in (0, 2/\lambda_{\max}(P_{\Omega}(\tilde{\mathbf{H}})^H P_{\Omega}(\tilde{\mathbf{H}})))$  according to [26], where  $\lambda_{\max}$  is the maximum eigenvalue;
- $\mu_{m+1}$  decreases as

$$\mu_{m+1} = \max\{\mu_m \eta_{\mu}, \mu_{\text{final}}\}, \quad m = 1, 2, \dots, M,$$

where  $M$ , determined by  $(\mu_{\text{final}}, \eta_{\mu})$ , controls the step size and the estimation accuracy,  $\mu_{\text{final}}$  is small (e.g.,  $\mu_{\text{final}} = 0.01$ ), and the parameter  $0 < \eta_{\mu} < 1$  determines the decreasing rate for consecutive  $\mu_m$ .

---

#### Algorithm 2 FPC Estimator

---

**Require:**  $P_{\Omega}(\tilde{\mathbf{H}})$ ,  $\epsilon$ ,  $J_{\text{FPC}}$ ,  $\delta$ ,  $\mu_{\text{final}}$ , and  $\eta_{\mu}$

```

1: Initialization:  $\hat{\mathbf{H}}^0 = \mathbf{0}$ ,  $m = 0$ ,  $\mu_m = \|P_{\Omega}(\tilde{\mathbf{H}})\|_2$ 
2: while  $\mu_m > \mu_{\text{final}}$  do
3:    $\mu_m = \max(\mu_m \eta_{\mu}, \mu_{\text{final}})$ 
4:   for  $k = 1 : J_{\text{FPC}}$  do
5:      $\mathbf{Y}^k = \hat{\mathbf{H}}^k - \delta(P_{\Omega}(\tilde{\mathbf{H}}) - P_{\Omega}(\hat{\mathbf{H}}^k))$ 
6:      $\hat{\mathbf{H}}^k = \mathcal{S}_{\delta\mu_m}(\mathbf{Y}^k)$ 
7:     if  $\frac{\|\hat{\mathbf{H}}^{k+1} - \hat{\mathbf{H}}^k\|_{\mathbb{F}}}{\|\hat{\mathbf{H}}^k\|_{\mathbb{F}}} < \epsilon$  then break;
8:     end if
9:   end for
10: end while
11: return  $\hat{\mathbf{H}} = \hat{\mathbf{H}}^k$ 

```

---

The main computational cost of the FPC algorithm is in Step 6 of Algorithm 2 due to the SVD. In addition, the FPC algorithm needs to choose the step size  $\delta$  by calculating the maximum eigenvalue of  $P_{\Omega}(\tilde{\mathbf{H}})^H P_{\Omega}(\tilde{\mathbf{H}})$  and has a higher computational complexity per iteration than that of the SVT algorithm.

#### 4.1.3. SVP Estimator

The SVP algorithm [22] is based on the projected gradients and is detailed in Algorithm 3. This algorithm requires that the rank  $L$  of the channel matrix to be known. The step size can be chosen empirically as  $\eta = 1/(1 + \delta_0)p$  with  $0 < \delta_0 < 1/3$ . The stopping criterion is based on the norm of the difference in the sampled channel matrix, where the small threshold  $0 < \epsilon < 1/2$  can be set such as  $\epsilon = 10^{-3}$ . The SVP algorithm also needs to calculate the SVD in Step 4, which is the most computationally expensive step of the algorithm.

**Algorithm 3** SVP Estimator

---

**Require:**  $P_{\Omega}(\tilde{\mathbf{H}})$ ,  $L$ ,  $\eta$ ,  $\epsilon$

- 1: **Initialization:**  $\hat{\mathbf{H}}^0 = \mathbf{0}$ ,  $t = 0$
- 2: **while**  $\|P_{\Omega}(\hat{\mathbf{H}}^t - \tilde{\mathbf{H}})\|_{\text{F}} \leq \epsilon$  **do**
- 3:    $\mathbf{X}^{t+1} \leftarrow \hat{\mathbf{H}}^t - \eta(P_{\Omega}(\hat{\mathbf{H}}^t - \tilde{\mathbf{H}}))$
- 4:   Compute the  $L$  principal singular vectors of  $\mathbf{X}^{t+1}$ :  $\mathbf{U}_L, \Sigma_L, \mathbf{V}_L$ .
- 5:    $\hat{\mathbf{H}}^{t+1} = \mathbf{U}_L \Sigma_L \mathbf{V}_L^H$
- 6:    $t = t + 1$ ;
- 7: **end while**
- 8: **return**  $\hat{\mathbf{H}} = \hat{\mathbf{H}}^k$

---

## 4.1.4. GCG-Alt Estimator

In [20], a generalized conditional gradient framework with alternating minimization (GCG-Alt) is developed for the MC problem. The nuclear norm is used to promote low-rankness as:

$$\min_{\hat{\mathbf{H}}} \frac{1}{2} \|P_{\Omega}(\hat{\mathbf{H}} - \tilde{\mathbf{H}})\|_{\text{F}}^2 + \mu \|\hat{\mathbf{H}}\|_* \quad (15)$$

where  $\mu > 0$  is a regularization factor. Let:

$$f(\hat{\mathbf{H}}) \triangleq \frac{1}{2} \|P_{\Omega}(\hat{\mathbf{H}}) - P_{\Omega}(\tilde{\mathbf{H}})\|_{\text{F}}^2. \quad (16)$$

The gradient direction of the  $k$ th iteration of  $f(\hat{\mathbf{H}})$  [23]:

$$\mathbf{X}_k = \mathbf{u}_{k-1} \mathbf{v}_{k-1}^H, \quad (17)$$

where  $(\mathbf{u}_{k-1}, \mathbf{v}_{k-1})$  is the top singular vector pair of  $\nabla f(\hat{\mathbf{H}}_{k-1}) = P_{\Omega}(\hat{\mathbf{H}}_{k-1} - \tilde{\mathbf{H}})$  which can be found iteratively. The channel matrix is updated as:

$$\hat{\mathbf{H}}_k = (1 - \eta_k) \hat{\mathbf{H}}_k + \theta_k \mathbf{X}_k, \quad (18)$$

where  $\eta_k \in [0, 1]$  is the step size and  $\theta_k$  is adaptively chosen.

By using a property of nuclear norm [20], the optimization problem can be reformulated as:

$$\tilde{\phi}(\mathbf{U}, \mathbf{V}) \triangleq f(\mathbf{U}\mathbf{V}^H) + \frac{1}{2} \mu (\|\mathbf{U}\|_{\text{F}}^2 + \|\mathbf{V}\|_{\text{F}}^2), \quad (19)$$

where  $\mathbf{U} \in \mathbb{C}^{N_{\text{MS}} \times \hat{r}}$  and  $\mathbf{V} \in \mathbb{C}^{N_{\text{BS}} \times \hat{r}}$  with  $\hat{r}$  being the rank of  $\hat{\mathbf{H}}$ . Alternating minimization can then be used to optimize Equation (19). The details of solving the alternate minimization problem can be found in [20]. The overall algorithm is summarized in Algorithm 4.

**Algorithm 4** GCG-Alt Estimator**Require:**  $P_{\Omega}(\tilde{\mathbf{H}}), \mu, \epsilon, \epsilon_a$ 


---

```

1: Initialization:  $\mathbf{U}_0 = \emptyset, \mathbf{V}_0 = \emptyset, k = 0, \hat{\mathbf{H}}_k = \mathbf{0}, \epsilon_0 = \infty, \delta_k^2 = \left\| P_{\Omega}(\tilde{\mathbf{H}}) \right\|_F^2$ 
2: while  $\delta_k^2 > (N + \sqrt{8N})\sigma^2$  do
3:   Compute the top singular vector pair  $(\mathbf{u}_k, \mathbf{v}_k)$  of  $P_{\Omega}(\hat{\mathbf{H}}_k - \tilde{\mathbf{H}})$ 
4:    $k = k + 1$ 
5:    $\eta_k \leftarrow 2/(k + 1)$ 
6:    $\mathbf{x}_{k\Omega} = \text{vec}(P_{\Omega}(\mathbf{u}_k \mathbf{v}_k^H))$ 
7:    $\tilde{\mathbf{h}}_{\Omega} = \text{vec}(P_{\Omega}(\tilde{\mathbf{H}}))$ 
8:    $\hat{\mathbf{h}}_{k\Omega} = \text{vec}(P_{\Omega}(\hat{\mathbf{H}}_{k-1}))$ 
9:    $\theta_k = \frac{2\mathcal{R}(\mathbf{x}_{k\Omega}^H \tilde{\mathbf{h}}_{\Omega}) - (1 - \eta_k) \mathbf{x}_{k\Omega}^H \hat{\mathbf{h}}_{k\Omega} - 2\mu}{2\mathbf{x}_{k\Omega}^H \mathbf{x}_{k\Omega}}$ 
10:   $\mathbf{U}_k = [\sqrt{1 - \eta_k} \mathbf{U}_{k-1}, \sqrt{\theta_k} \mathbf{u}_{k-1}]$ 
11:   $\mathbf{V}_k = [\sqrt{1 - \eta_k} \mathbf{V}_{k-1}, \sqrt{\theta_k} \mathbf{v}_{k-1}]$ 
12:  Initialization:  $i = 0, \epsilon_k^0 = \infty, (\mathbf{U}_k^0, \mathbf{V}_k^0) \leftarrow (\mathbf{U}_k, \mathbf{V}_k)$ .
13:  while  $\epsilon_k^i > \epsilon_a$  do
14:     $i = i + 1$ 
15:    Find  $\mathbf{V}_k^i$  that minimizes Equation (19) given  $\mathbf{U} = \mathbf{U}_k^i$ ;
16:    Find  $\mathbf{U}_k^i$  that minimizes Equation (19) given  $\mathbf{V} = \mathbf{V}_k^i$ ;
17:    Calculate
        
$$\epsilon_k^i = \frac{\tilde{\phi}(\mathbf{U}_k^{i-1}, \mathbf{V}_k^{i-1}) - \tilde{\phi}(\mathbf{U}_k^i, \mathbf{V}_k^i)}{\tilde{\phi}(\mathbf{U}_k^{i-1}, \mathbf{V}_k^{i-1})}$$

18:  end while
19:   $(\mathbf{U}_k, \mathbf{V}_k) \leftarrow (\mathbf{U}_k^i, \mathbf{V}_k^i)$ 
20:  Calculate  $\epsilon_k = \frac{\|\hat{\mathbf{H}}_k\|_F^2 - \|\hat{\mathbf{H}}_{k-1}\|_F^2}{\|\hat{\mathbf{H}}_{k-1}\|_F^2}$ 
21:   $\delta_k^2 = \left\| P_{\Omega}(\hat{\mathbf{H}}_k - \tilde{\mathbf{H}}) \right\|_F^2$ 
22: end while
23: return  $\hat{\mathbf{H}} = \mathbf{U}_k \mathbf{V}_k^H$ 

```

---

The above MC methods have different computational complexities. SVT, SVP, and FPC all need SVD, which can be implemented using the PROPACK [35] based on the iterative Lanczos bidiagonalization algorithm with partial reorthogonalization. The FPC has a higher complexity as SVD is repeated for different values of  $\mu_m$ . The GCG-Alt has the least complexity as the full SVD is not required. SVP is effective for large matrix completion problems with very low ranks, while the FPC, SVT, and GCG-Alt estimators allow higher ranks. The SVP estimator requires rank knowledge, while the FPC and GCG-Alt estimators implicitly determine the rank through choosing regularization parameters or thresholds.

#### 4.2. MC-Based Hybrid Estimators

Next we discuss two MC-based hybrid methods that jointly exploit the sparsity and low-rankness of the channel.



#### 4.2.1. ADMM Estimator

In [17], the low-rankness of  $\tilde{\mathbf{H}}$  and the sparsity of the beamspace channel  $\tilde{\mathbf{H}}_v$  are jointly exploited and an ADMM method is proposed. Leveraging the side information that  $\mathbf{H}$  has a sparse virtual representation given by Equation (5), the channel estimation problem is formulated following [36] as:

$$\begin{aligned} \min_{\hat{\mathbf{H}}, \hat{\mathbf{H}}_v} \quad & \tau_L \|\hat{\mathbf{H}}\|_* + \tau_S \|\hat{\mathbf{H}}_v\|_1 \\ \text{s.t.} \quad & P_\Omega(\hat{\mathbf{H}}) = P_\Omega(\tilde{\mathbf{H}}) \text{ and } \hat{\mathbf{H}} = \mathbf{A}_{\text{MS}} \hat{\mathbf{H}}_v \mathbf{A}_{\text{BS}}^H \end{aligned} \quad (20)$$

where the nuclear norm and  $l_1$ -norm together with the regularization parameters  $\tau_L$  and  $\tau_S$  are used to promote low-rankness and sparsity, respectively. The above problem is then reformulated by incorporating the constraints as penalty terms as:

$$\begin{aligned} \min_{\hat{\mathbf{H}}, \mathbf{E}, \hat{\mathbf{H}}_v, \mathbf{C}} \quad & \tau_L \|\hat{\mathbf{H}}\|_* + \tau_S \|\hat{\mathbf{H}}_v\|_1 + \frac{1}{2} \|\mathbf{C}\|_F^2 + \frac{1}{2} \|P_\Omega(\mathbf{E} - \hat{\mathbf{H}})\|_F^2 \\ \text{s.t.} \quad & \hat{\mathbf{H}} = \mathbf{E} \text{ and } \mathbf{C} = \mathbf{E} - \mathbf{A}_{\text{MS}} \hat{\mathbf{H}}_v \mathbf{A}_{\text{BS}}^H. \end{aligned} \quad (21)$$

where  $\mathbf{E} \in \mathbb{C}^{N_{\text{MS}} \times N_{\text{BS}}}$  and  $\mathbf{C}$  are two auxiliary matrix variables. This problem is then solved by using ADMM which involves the iterative updates of the variables and Lagrangian multipliers. The augmented Lagrangian function of Equation (21) is given by:

$$\begin{aligned} \mathcal{L}_1(\hat{\mathbf{H}}, \mathbf{E}, \hat{\mathbf{H}}_v, \mathbf{C}, \mathbf{Z}_1, \mathbf{Z}_2) \triangleq & \tau_L \|\hat{\mathbf{H}}\|_* + \tau_S \|\hat{\mathbf{H}}_v\|_1 + \frac{1}{2} \|\mathbf{C}\|_F^2 \\ & + \frac{1}{2} \|P_\Omega(\mathbf{E} - \hat{\mathbf{H}})\|_F^2 + \text{tr}(\mathbf{Z}_1^H (\hat{\mathbf{H}} - \mathbf{E})) + \frac{t}{2} \|\hat{\mathbf{H}} - \mathbf{E}\|_F^2 \\ & + \text{tr}(\mathbf{Z}_2^H (\mathbf{E} - \mathbf{A}_{\text{MS}} \hat{\mathbf{H}}_v \mathbf{A}_{\text{BS}}^H - \mathbf{C})) \\ & + \frac{t}{2} \|\mathbf{E} - \mathbf{A}_{\text{MS}} \hat{\mathbf{H}}_v \mathbf{A}_{\text{BS}}^H - \mathbf{C}\|_F^2, \end{aligned} \quad (22)$$

where  $\mathbf{Z}_1$  and  $\mathbf{Z}_2 \in \mathbb{C}^{N_{\text{MS}} \times N_{\text{BS}}}$  are dual variables (the Lagrange multipliers) and  $t > 0$  is the step size. The estimator is summarized in Algorithm 5, where:

- $\tau = \rho_1 \|P_\Omega(\tilde{\mathbf{H}})\|$  with  $\rho_1 = \frac{3N}{N_{\text{BS}}N_{\text{MS}}}$  in Step 3;
- $\mathbf{z}_i$  denotes the vectorization of  $\mathbf{Z}_i$ , and similarly for other variables;
- $\mathbf{A} \triangleq \sum_{i=1}^{N_{\text{MS}}} \text{diag}([\mathbf{\Omega}_*]_i)^T \otimes \mathbf{I}_i$  where  $\mathbf{\Omega}_* \in \{0, 1\}^{N_{\text{MS}} \times N_{\text{BS}}}$  is composed of  $N$  ones and  $N_{\text{BS}}N_{\text{MS}} - N$  zeros, the value 1 indicates the position of a sample from the channel matrix, and  $[\mathbf{\Omega}_*]_i$  denotes the  $i$ -th row of  $\mathbf{\Omega}_*$ , and  $\mathbf{I}$  is the  $N_{\text{MS}} \times N_{\text{MS}}$  matrix that the value at its  $(i, i)$ -th position is 1 and the remaining position is 0 [17];
- The parameters in Equation (20) are chosen empirically as  $\tau_L = t \|P_\Omega(\tilde{\mathbf{H}})\|_2$  and  $\tau_S = \frac{0.1}{1 - 10 \log(\sigma_n^2)}$ , where  $\sigma_n^2$  is the noise power.

The computational cost of the ADMM algorithm is mainly due to the SVD in Step 3 and the matrix operations in Steps 4, 5, 6, and 7.

**Algorithm 5** ADMM Estimator**Require:**  $P_{\Omega}(\tilde{\mathbf{H}})$ ,  $t$ ,  $\tau_L$ ,  $\tau_S$ ,  $J_{\text{ADMM}}$ ,  $\mathbf{A}$ ,  $\Psi$ ,  $\tau$ .

- 1: **Initialization:**  $\hat{\mathbf{H}}^0 = \hat{\mathbf{H}}_v^0 = \mathbf{C}^0 = \mathbf{E}^0 = \mathbf{Z}_2^0 = \mathbf{Z}_1^0 = \mathbf{0}$ ; their vectorizations  $\hat{\mathbf{h}}^0 = \hat{\mathbf{h}}_v^0 = \mathbf{c}^0 = \mathbf{e}^0 = \mathbf{z}_2^0 = \mathbf{z}_1^0 = \mathbf{0}$ ;
- 2: **for**  $\ell = 0, 1, \dots, J_{\text{ADMM}} - 1$  **do**
- 3:     Update  $\hat{\mathbf{H}}^{\ell+1} = \mathbf{U}_A \mathcal{S}_{\tau}(\mathbf{S}_A) \mathbf{V}_A^H$ , where  $[\mathbf{U}_A, \mathbf{S}_A, \mathbf{V}_A] = \mathbf{E}^{\ell} - \frac{1}{t} \mathbf{Z}_1^{\ell}$
- 4:     Update

$$\mathbf{e}^{\ell+1} = \left( \mathbf{A}^H \mathbf{A} + 2t\mathbf{I} \right)^{-1} \left( \mathbf{z}_1^{\ell} + t\hat{\mathbf{h}}^{\ell+1} + \mathbf{A}^H \text{vec}(P_{\Omega}(\tilde{\mathbf{H}})) + \mathbf{z}_2^{\ell} + t\mathbf{c}^{\ell} + t\Psi\hat{\mathbf{h}}_v^{\ell} \right).$$

- 5:     Update

$$\begin{aligned} \hat{\mathbf{h}}_v^{\ell+1} = & \text{sign} \left( \mathcal{R} \left( \mathbf{v}^{\ell+1} \right) \right) \circ \max \left( \left| \mathcal{R} \left( \mathbf{v}^{\ell+1} \right) \right| - \tau'_S, 0 \right) \\ & + \text{sign} \left( \mathcal{I} \left( \mathbf{v}^{\ell+1} \right) \right) \circ \max \left( \left| \mathcal{I} \left( \mathbf{v}^{\ell+1} \right) \right| - \tau'_S, 0 \right) \end{aligned}$$

with  $\mathbf{V}^{\ell+1} \triangleq \mathbf{A}_{\text{MS}}^H \left( \frac{1}{t} \mathbf{Z}_2^{\ell} - \mathbf{C}^{\ell} + \mathbf{E}^{\ell+1} \right) \mathbf{A}_{\text{BS}}$  and  $\tau'_S \triangleq \tau_S / t$ .

- 6:     Update

$$\mathbf{C}^{\ell+1} = \frac{t}{t+1} \left( \mathbf{E}^{\ell+1} - \mathbf{A}_{\text{MS}} \hat{\mathbf{H}}_v^{\ell+1} \mathbf{A}_{\text{BS}}^H + \frac{1}{t} \mathbf{Z}_2^{\ell} \right)$$

- 7:     Update

$$\begin{aligned} \mathbf{Z}_1^{\ell+1} &= \mathbf{Z}_1^{\ell} + t \left( P_{\Omega}(\mathbf{E}^{\ell+1} - \tilde{\mathbf{H}}) \right) \\ \mathbf{Z}_2^{\ell+1} &= \mathbf{Z}_2^{\ell} + t \left( \mathbf{E}^{\ell+1} - \mathbf{A}_{\text{MS}} \hat{\mathbf{H}}_v^{\ell+1} \mathbf{A}_{\text{BS}}^H - \mathbf{C}^{\ell+1} \right) \end{aligned}$$

- 8: **end for;**

- 9: **return**  $\hat{\mathbf{H}} = \hat{\mathbf{H}}^{J_{\text{ADMM}}-1}$

## 4.2.2. Two-Stage Estimator

The two-stage estimator [16] also exploits the sparsity and low-rankness of the channel matrix. In the first stage, MC is applied to provide a denoised channel estimation based on the low rankness of the channel matrix and then the second stage employs CS to refine the estimation based on the array response and the virtual representation of the channel matrix.

The SVT algorithm [25] introduced above solves the low-rank matrix completion problem:

$$\min_{\hat{\mathbf{H}}} \text{rank}(\hat{\mathbf{H}}), \quad \text{s.t.} \quad \|P_{\Omega}(\hat{\mathbf{H}}) - P_{\Omega}(\tilde{\mathbf{H}})\|_F^2 \leq \delta_n^2. \quad (23)$$

The fast iterative shrinkage threshold algorithm (FISTA) proposed by [37] is used to solve the sparse vector recovery problem:

$$\min_{\text{vec}(\hat{\mathbf{H}}_v)} \|\Psi \text{vec}(\hat{\mathbf{H}}_v) - \text{vec}(\hat{\mathbf{H}})\|_F^2 + \lambda \|\text{vec}(\hat{\mathbf{H}}_v)\|_1. \quad (24)$$

The estimator is summarized in Algorithm 6, where:

- The parameters of the first stage, SVT, is the same with Algorithm 1;
- $\Psi = \mathbf{A}_{\text{BS}}^* \otimes \mathbf{A}_{\text{MS}}$ ;
- $\lambda$  is a constant stepsize, e.g.,  $\lambda = 0.001$ ;
- $\lambda_{\text{max}}$  is the top eigenvalue of  $\Psi^H \Psi$ .

The complexity of SVT in the first stage has been analyzed. The cost of the FISTA algorithm mainly consists of applying the sensing matrix in Step 4, which has a complexity of  $O(N_{\text{BS}}^2 N_{\text{MS}}^2)$ .

**Algorithm 6** Two-Stage Estimator**Require:**  $P_{\Omega}(\tilde{\mathbf{H}})$ 1: Use SVT to recover  $\hat{\mathbf{H}}$  as

$$\min_{\hat{\mathbf{H}}} \|\hat{\mathbf{H}}\|_* \text{ s.t. } \|P_{\Omega}(\hat{\mathbf{H}}) - P_{\Omega}(\tilde{\mathbf{H}})\|_F^2 \leq \delta_n^2$$

**Require:**  $\Psi$ ,  $\text{vec}(\hat{\mathbf{H}})$ ,  $\lambda$ ,  $J_{\text{FISTA}}$ , the top eigenvalue  $\lambda_{\max}$  of  $\Psi^H\Psi$ 2: Initialize  $\mathbf{y}_1 = \mathbf{x}_1 = \mathbf{0}$ ,  $t_1 = 1$ 3: **for**  $i = 1 : J_{\text{FISTA}}$  **do**4:    $\mathbf{c}_i = \mathbf{y}_i - (1/\lambda_{\max})\Psi^H(\Psi\mathbf{y}_i - \text{vec}(\hat{\mathbf{H}}))$ 5:    $\mathbf{x}_{i+1} = (\max(\text{abs}(\mathbf{c}_i) - 2\lambda\lambda_{\max}, 0)) \text{sign}(\mathbf{c}_i)$ 6:    $t_{i+1} = \frac{1 + \sqrt{1 + 4t_i^2}}{2}$ 7:    $\mathbf{y}_{i+1} = \mathbf{x}_{i+1} + \left(\frac{t_i - 1}{t_{i+1}}\right) (\mathbf{x}_{i+1} - \mathbf{x}_i)$ 8:    $\mathbf{x}_i \leftarrow \mathbf{x}_{i+1}$ 9:    $\epsilon_{i+1} = \|\Psi\mathbf{x}_i - \text{vec}(\hat{\mathbf{H}})\|_2$ 10:   **if**  $\text{abs}(\epsilon_{i+1} - \epsilon_i) \leq 10^{-6}$  **then**

11:     Break

12:   **end if**13: **end for**14: **return**  $\hat{\mathbf{H}} = \mathbf{A}_{\text{MS}} \text{vec}^{-1}(\mathbf{x}_i) \mathbf{A}_{\text{BS}}^H$ **5. Numerical Results**

Consider switch-based MIMO systems over a mmWave channel at 90 GHz. When not otherwise specified, the number of clusters  $C \sim \max(\text{Poisson}(1.8), 1)$ , and the number of rays  $R \sim \mathcal{U}[1, 20]$  (the total number of paths is  $L = CR$ ); the AoDs and AoAs follow Laplace distributions with a standard deviation of  $15^\circ$  [38]; the sub-power of the clusters  $\gamma_c = 1$ ; and the ULA at the MS and BS has  $N_{\text{MS}} = 32$  antennas and  $N_{\text{BS}} = 128$  antennas, respectively, and  $N_{\text{RF}} = 4$  RF chains. Here, the CS method based on OMP is compared with MC-based estimators. The parameter settings are as follows:

- OMP: The unitary dictionary is set with  $N_1 = N_{\text{MS}} = 32$  and  $N_2 = N_{\text{BS}} = 128$ . The stopping threshold is set as  $\epsilon_{\text{OMP}} = 0.025\sigma^2, 0.05\sigma^2, 0.075\sigma^2, 0.1\sigma^2, 0.125\sigma^2, 0.15\sigma^2$  with  $\text{PNR} = 0, 5, 10, 15, 20, 25$  dB [20];
- SVT:  $\delta = 3.2$ ,  $\epsilon = 10^{-4}$ ,  $\tau = 5\sqrt{N_{\text{MS}}N_{\text{BS}}}$ ,  $J_{\text{SVT}} = 100$ , and  $k_0 = 5$ ;
- SVP:  $\eta = 0.5$  and  $\epsilon = 10^{-4}$ ;
- FPC:  $\mu_{\text{final}} = 0.01$ ,  $J_{\text{FPC}} = 100$ ,  $\epsilon = 10^{-4}$ ,  $\eta_{\mu} = 0.25$ , and  $\delta = 1.99$ ;
- GCG-Alt:  $\mu = \sigma_n^2$ ,  $\epsilon = 0.01$ , and  $\epsilon_a = 0.1$ ;
- ADMM:  $\tau_L = t \|P_{\Omega}(\tilde{\mathbf{H}})\|_2$ ,  $t = 0.005$ ,  $\tau_S = \frac{0.1}{1 - 10 \log(\sigma_n^2)}$ ,  $\delta = 3.2$ ;
- Two-Stage:  $J_{\text{FISTA}} = 100$ ,  $\lambda = 0.001$ .

The normalized MSE (NMSE) is defined as:

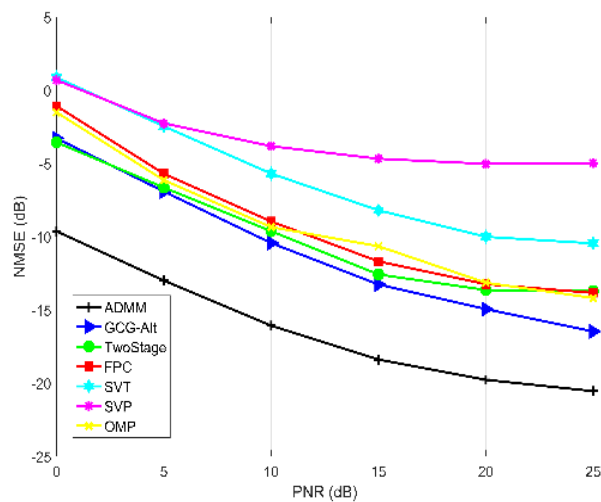
$$\text{NMSE} = \frac{\|\hat{\mathbf{H}} - \mathbf{H}\|_F^2}{\|\mathbf{H}\|_F^2}$$

where  $\hat{\mathbf{H}}$  is an estimate of the channel matrix  $\mathbf{H}$ .

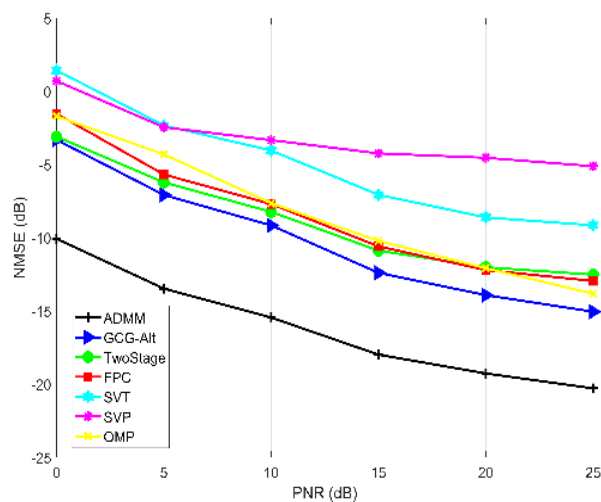
**5.1. Comparison of NMSE When There Are No Hardware Impairments**

Figure 2 compares different estimators in terms of NMSE with different PNRs, which is defined as  $\text{PNR} = 10 \log 10(\frac{\rho}{\sigma_n^2})$ , when  $N_{\text{MS}} = 32$ ,  $N_{\text{BS}} = 128$ , and  $N_{\text{RF}} = 4$ . The SVP algorithm performed worse than others. The ADMM algorithm performed the best, and the NMSE of the ADMM estimator was at least 5 dB better than other estimators. The NMSE with  $N_{\text{BS}} = 64$ ,  $N_{\text{MS}} = 64$  and  $N_{\text{RF}} = 8$  is

shown in Figure 3. From Figures 2 and 3, the relative performance of the different estimators is similar with different numbers of antennas.

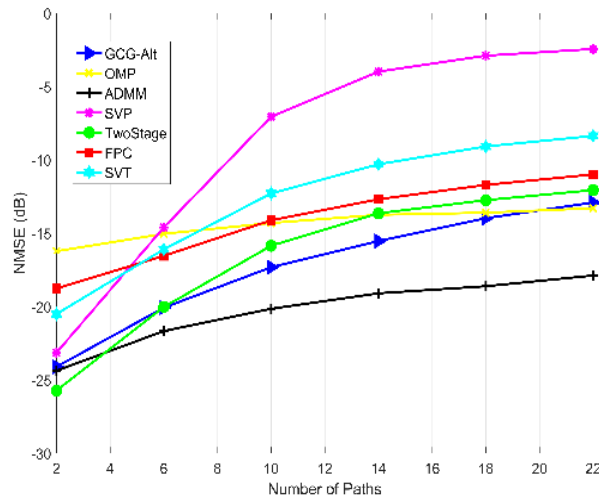


**Figure 2.** NMSE (normalized mean square error) of the channel estimation in the ULA (uniform linear arrays) system with  $N_{BS} = 128$ ,  $N_{MS} = 32$ ,  $N_{RF} = 4$ , the sampling ratio  $p = 0.375$ , different PNRs (pilot-to-noise ratios), and without array impairments.

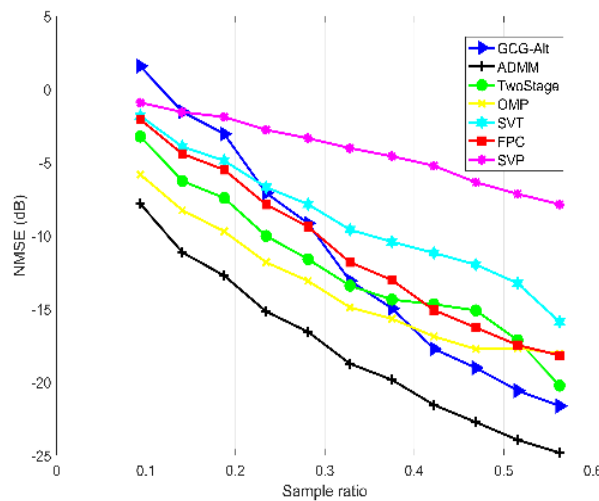


**Figure 3.** NMSE of the channel estimation in the ULA system with  $N_{BS} = 64$ ,  $N_{MS} = 64$ ,  $N_{RF} = 8$ ,  $p = 0.3281$ , different PNRs, and without array impairments.

We next show the performance with different number of channel paths. Here, we assume the number of clusters  $C = 1$ , the numbers of rays  $R$  is changed from 1 to 22, and so the total number of channel paths  $L = R$ . From Figure 4, as the numbers of paths  $L$  increases, the NMSE of different algorithms degrades. The SVP algorithm is the most sensitive to the number of paths while the OMP estimator is the most robust. The sampling ratio is also critical for performance. From Figure 5, the NMSE improves substantially with increasing sampling ratio. It is noted that CS-based schemes such as the OMP and the two-stage algorithms are more advantageous when the sampling ratio is low.



**Figure 4.** NMSE of the channel estimation in the ULA system with  $N_{BS} = 128$ ,  $N_{MS} = 32$ ,  $N_{RF} = 4$ ,  $p = 0.375$ , PNR = 20 dB, different number of paths, and without array impairments.



**Figure 5.** NMSE of the channel estimation in the ULA system with  $N_{BS} = 128$ ,  $N_{MS} = 32$ ,  $N_{RF} = 4$ , PNR = 20dB, different sample ratios, and without array impairments.

## 5.2. NMSE Comparison When There Are Hardware Impairments

In practice, it is inevitable to have impairments of the antenna elements, which are typically time-varying, e.g., due to temperature changes or hardware aging [39]. Therefore, the array response may be severely impacted. Due to mechanical reasons and uncertainty regarding the precise position of the antenna phase center, the actual antenna position may deviate from the assumed ideal array shape. Following [20], we define the gain and phase error vector at the BS as:

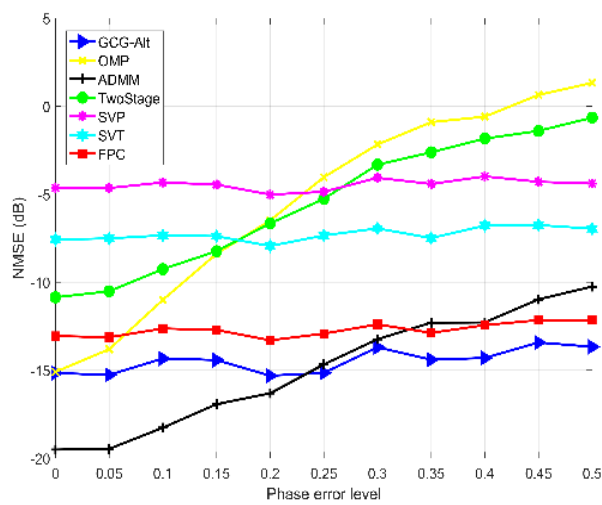
$$\mathbf{e}_{BS} = [\beta_1 e^{j\omega_1}, \beta_2 e^{j\omega_2}, \dots, \beta_{N_{BS}} e^{j\omega_{N_{BS}}}]^T \quad (25)$$

where  $\omega_i$  represents the phase errors and  $\beta_i$  denotes the amplitude gain of each antenna element. The gain and phase error vector  $\mathbf{e}_{MS}$  at the MS is similar to  $\mathbf{e}_{BS}$ . Considering such errors, the received signal in Equation (2) can be expressed as:

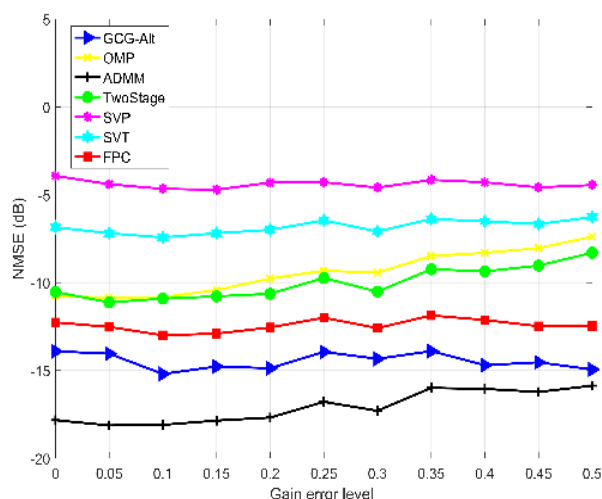
$$\tilde{\mathbf{y}} = \mathbf{W}^H \mathbf{E}_{MS} \mathbf{H} \mathbf{E}_{BS}^H \mathbf{F}_S + \mathbf{W}^H \mathbf{E}_{MS} \hat{\mathbf{n}} \quad (26)$$

where  $\mathbf{E}_{MS}$  is a diagonal matrix with  $\mathbf{e}_{MS}$  as the diagonal elements, and  $\mathbf{E}_{BS}$  is defined similarly.

We carry out simulations to examine the robustness of the different approaches when there are phase and gain errors in the array response. Those errors are assumed to be uniformly distributed within certain range and characterized by the level of phase and gain errors, respectively, following [20]. It is found that MC estimators were not affected by the phase or gain errors since the estimators were independent of the array response vectors. The sparsity-based methods use array response which depends on the phase and gain information, and can result in poor channel estimation when unknown gain and phase errors are present. These were validated by simulation experiments. From Figures 6 and 7 where NMSE achieved with different levels of phase and gain errors are compared, the MC estimators based on SVP, FPC, GCG-Alt, and SVT, were insensitive to phase errors or gain errors. The estimators exploiting CS, such as the one based on OMP and hybrid methods (the ADMM and two-stage estimators), were more sensitive to phase errors or gain errors.



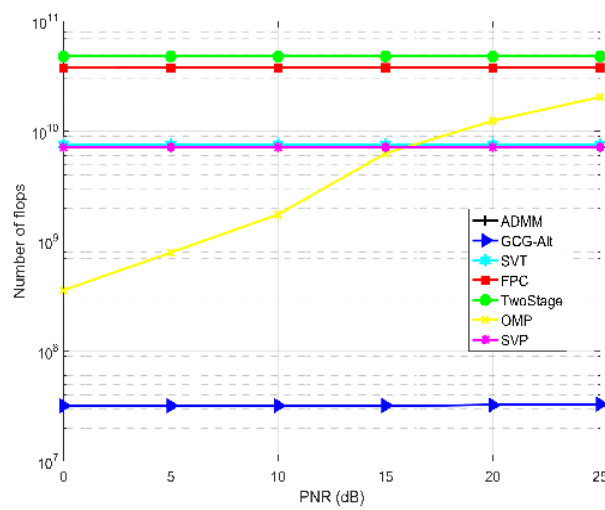
**Figure 6.** NMSE of the channel estimation in the ULA system with  $N_{BS} = 128$ ,  $N_{MS} = 32$ ,  $N_{RF} = 4$ ,  $p = 0.375$ , PNR = 20 dB, and different levels of phase errors.



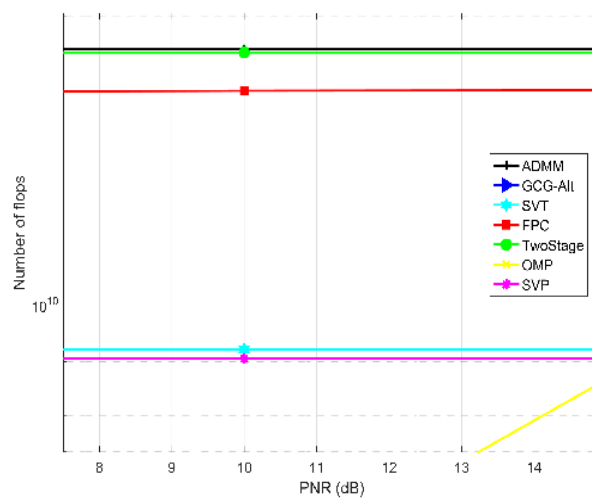
**Figure 7.** NMSE of the channel estimation in the ULA system with  $N_{BS} = 128$ ,  $N_{MS} = 32$ ,  $N_{RF} = 4$ ,  $p = 0.375$ , PNR = 20 dB, and different levels of gain errors.

### 5.3. Computational Complexity

Finally, we compare the computational complexity of different algorithms in Figures 8 and 9. It can be seen that in general, the hybrid estimators (the ADMM and two-stage estimators) exhibited higher complexity because they both involved SVD and the application of the sensing matrices were of large sizes. In particular, the ADMM algorithm had the highest complexity. The MC estimators had moderate complexity that did not vary significantly with the PNRs and the FPC exhibited higher complexity than SVT and SVP as more SVD operations were required. The GCG-Alt algorithm had the lowest complexity. This was because the GCG-Alt algorithm had a fast convergence rate and also avoided the SVD operations used in the other MC algorithms such as SVP, SVT, and FPC. The computational complexity of the OMP estimator increased with the PNR because the numbers of paths recovered by the OMP increased as the PNR increased.



**Figure 8.** Complexity comparison for the ULA system with different PNRs  $N_{BS} = 128$ ,  $N_{MS} = 32$ ,  $N_{RF} = 4$ , and  $p = 0.375$ .



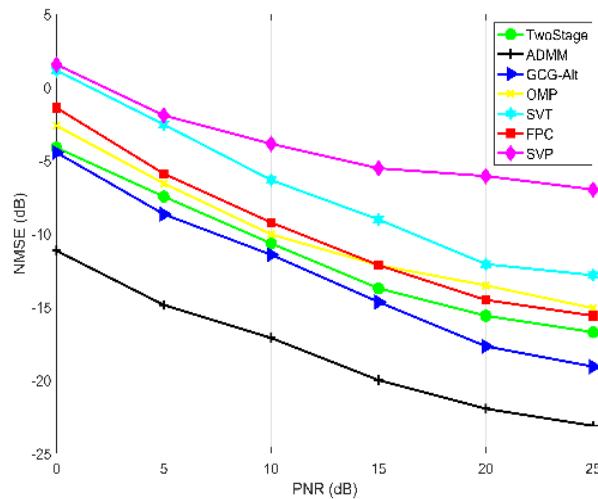
**Figure 9.** Zoomed-in section of Figure 8 with different PNRs  $N_{BS} = 128$ ,  $N_{MS} = 32$ ,  $N_{RF} = 4$ , and  $p = 0.375$ .

#### 5.4. Performance with Line of Sight (LoS) Propagation

In the above, we have focused on the comparison for the channels where the different paths had the same average power. In practical applications, there may exist line of sight (LoS) propagation, where a path contributes a significant portion of the power gain [40,41]. In this case, the channel model can be modified from Equation (3) to:

$$\mathbf{H} = \frac{1}{\sqrt{R}} \sum_{c=1}^C \sum_{r=1}^R g_{cr} \mathbf{a}_{\text{MS}}(\phi_{cr}^{\text{MS}}) \mathbf{a}_{\text{BS}}^{\text{H}}(\phi_{cr}^{\text{BS}}) + \beta \mathbf{a}_{\text{MS}}(\phi_{\text{LoS}}^{\text{MS}}) \mathbf{a}_{\text{BS}}^{\text{H}}(\phi_{\text{LoS}}^{\text{BS}}). \quad (27)$$

In our simulations we set  $g_{cr} \sim \mathcal{CN}(0, 0.5)$  and  $\beta = \sqrt{C/2}$ . The NMSE results are shown in Figure 10. With a LoS path which dominates the power gain, the effective rank of the channel matrix may be reduced. The various channel estimators considered are more effective in this case, leading to lower NMSE as compared with the case where the LoS path is absent. The results also demonstrate the effectiveness of the MC-based methods in realistic scenarios.



**Figure 10.** NMSE of the channel estimation in the ULA system with  $N_{\text{BS}} = 128$ ,  $N_{\text{MS}} = 32$ ,  $N_{\text{RF}} = 4$ , the sampling ratio  $p = 0.375$ , different PNRs, and without array impairments.

## 6. Conclusions

In this paper, we compared the performance of several MC-based channel estimators for mmWave massive MIMO systems. It was observed that the hybrid ADMM algorithm exhibited the best performance in general, which jointly exploited the low rank property of the channel matrix and the sparsity in the angular domain. However, it also exhibited the highest complexity among the estimators compared. The MC-based estimators (using GCG-Alt, SVT, SVP, or FPC) were robust against array impairments as they did not rely on array response vectors. Among them the GCG-Alt estimator exhibited the lowest complexity, better performance, and provided a competitive solution when the arrays were not perfectly calibrated.

In this work, we considered a point-to-point mmWave system. The estimators could also be applied to multiuser systems when orthogonal training schemes are deployed. The comparison of these methods when nonorthogonal training is used would be an interesting study for future work.

**Author Contributions:** Conceptualization, M.D., X.Y., and J.T.; Methodology, M.D. and X.Y.; Validation, R.H. and J.T.; Formal analysis, X.Y., R.H., and J.T.; Investigation, M.D., X.Y., and R.H.; Resources, M.D., Z.X. and J.X.; Data curation, X.Y. and R.H.; Writing—original draft preparation, X.Y. and R.H.; Writing—review and editing, Z.X., J.T., and J.X.; Visualization, X.Y.; Supervision, Z.X., J.T., and J.X.; Project administration, Z.X. and J.X.; Funding acquisition, Z.X., J.T., and J.X.



**Funding:** This research was funded by NSFC under Grant 61601325.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Rangan, S.; Rappaport, T.S.; Erkip, E. Millimeter-wave cellular wireless networks: Potentials and challenges. *Proc. IEEE* **2014**, *102*, 366–385. [[CrossRef](#)]
2. Rebeiz, G. Large-scale millimeter-wave phased arrays for 5g systems. In Proceedings of the IEEE 16th Topical Meeting Silicon Monolithic Integrated Circuits in RF Systems (SiRF), Austin, TX, USA, 24–27 January 2016.
3. Rappaport, T.S.; Heath, R.W.; Daniels, R.C.; Murdock, J. *Millimeter Wave Wireless Communications*; Prentice-Hall: Englewood Cliffs, NJ, USA, 2014.
4. Heath, R.W.; González-Prelcic, N.; Rangan, S.; Roh, W.; Sayeed, A.M. An Overview of Signal Processing Techniques for Millimeter Wave MIMO Systems. *IEEE J. Sel. Top. Signal Process.* **2016**, *10*, 436–453. [[CrossRef](#)]
5. Wei, L.; Hu, R.; Qian, Y.; Wu, G. Key elements to enable millimeter wave communications for 5g wireless systems. *IEEE Wirel. Commun.* **2014**, *21*, 136–143.
6. Eldar, Y.C.; Kutyniok, G. *Compressed Sensing*; Cambridge University Press: Cambridge, UK, 2009.
7. Lee, J.; Gil, G.T.; Lee, Y.H. Channel estimation via orthogonal matching pursuit for hybrid MIMO systems in millimeter wave communications. *IEEE Trans. Commun.* **2016**, *64*, 2370–2386. [[CrossRef](#)]
8. Alkhateeb, A.; Ayach, O.E.; Leus, G.; Heath, R.W. Channel estimation and hybrid precoding for millimeter wave cellular systems. *IEEE J. Sel. Top. Signal Process.* **2014**, *8*, 831–846. [[CrossRef](#)]
9. Gao, Z.; Dai, L.; Wang, Z.; Chen, S. Spatially common sparsity based adaptive channel estimation and feedback for FDD massive MIMO. *IEEE Trans. Signal Process.* **2015**, *63*, 6169–6183. [[CrossRef](#)]
10. Sun, S.; Rappaport, T.S. Millimeter wave MIMO channel estimation based on adaptive compressed sensing. In Proceedings of the 2017 IEEE International Conference on Communications Workshops (ICC Workshops), Paris, France, 21–25 May 2017; pp. 47–53.
11. Xiao, Z.; Yin, C.; Xia, P.; Xia, X. Codebook design for millimeter-wave channel estimation with hybrid precoding structure. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 141–153. [[CrossRef](#)]
12. Zhu, D.; Choi, J.; Heath, R.W. Auxiliary beam pair enabled AoD and AoA estimation in closed-loop large-scale millimeter-wave MIMO systems. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 4770–4785. [[CrossRef](#)]
13. Gurbuz, A.C.; Yapici, Y.; Guvenc, I. Sparse channel estimation in millimeter-wave communications via parameter perturbed OMP. In Proceedings of the 2018 IEEE International Conference on Communications Workshops (ICC Workshops), Kansas City, MO, USA, 20–24 May 2018.
14. Ayach, O.E.; Rajagopal, S.; Abu-Surra, S.; Pi, Z.; Heath, R.W. Spatially sparse precoding in millimeter wave MIMO systems. *IEEE Trans. Wirel. Commun.* **2014**, *13*, 1499–1513. [[CrossRef](#)]
15. Eliasi, P.A.; Rangan, S.; Rappaport, T.S. Low-rank spatial channel estimation for millimeter wave cellular systems. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 2748–2759. [[CrossRef](#)]
16. Li, X.; Fang, J.; Li, H.; Wang, P. Millimeter wave channel estimation via exploiting joint sparse and low-rank structures. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 1123–1133. [[CrossRef](#)]
17. Vlachos, E.; Alexandropoulos, G.C.; Thompson, J. Massive MIMO channel estimation for millimeter wave systems via matrix completion. *IEEE Signal Process. Lett.* **2018**, *25*, 1675–1679. [[CrossRef](#)]
18. Sun, J.; Lu, J.; Liang, G.; Bi, J. A sparse interactive model for matrix completion with side information. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 4071–4079.
19. Hu, R.; Tong, J.; Xi, J.; Guo, Q.; Yu, Y. Robust channel estimation for switch-based mmWave MIMO systems. In Proceedings of the 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 11–13 October 2017; pp. 1–7.
20. Hu, R.; Tong, J.; Xi, J.; Guo, Q.; Yu, Y. Matrix completion-based channel estimation for MmWave communication systems with array-inherent impairments. *IEEE Access* **2018**, *6*, 62915–62931. [[CrossRef](#)]
21. Akdeniz, M.R.; Liu, Y.; Samimi, M.K.; Sun, S.; Rangan, S.; Rappaport, T.S.; Erkip, E. Millimeter wave channel modeling and cellular capacity evaluation. *IEEE J. Sel. Areas Commun.* **2014**, *32*, 1164–1179. [[CrossRef](#)]
22. Jain, P.; Meka, R.; Dhillon, I.S. Guaranteed rank minimization via singular value projection. In Proceedings of the 23rd International Conference on Neural Information Processing Systems, Vancouver, BC, USA, 6–9 December 2010; pp. 937–945.

23. Yu, Y.; Carbonell, J.; Yu, A.W.; Ma, W.; Sra, S. Efficient structured matrix rank minimization. In Proceedings of the 28th Annual Conference on Neural Information Processing Systems (NIPS 2014), Montreal, QC, USA, 8–13 December 2014; pp. 1350–1358.
24. Yu, X.; Shen, J.C.; Zhang, J.; Letaief, K.B. Alternating minimization algorithms for hybrid precoding in millimeter wave MIMO systems. *IEEE J. Sel. Top. Signal Process.* **2016**, *10*, 485–500. [[CrossRef](#)]
25. Cai, J.F.; Candès, E.J.; Shen, Z. A singular value thresholding algorithm for matrix completion. *SIAM J. Optim.* **2010**, *20*, 1956–1982. [[CrossRef](#)]
26. Ma, S.; Goldfarb, D.; Chen, L. Fixed point and bregman iterative methods for matrix rank minimization. *Math. Program.* **2009**, *128*, 321–353. [[CrossRef](#)]
27. Xia, P.; Yong, S.K.; Oh, J.; Ngo, C. A practical SDMA protocol for 60 GHz millimeter wave communications. In Proceedings of the 2008 42nd Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 26–29 October 2008; pp. 2019–2023.
28. Forenza, A.; Love, D.J.; Heath, R.W. Simplified spatial correlation models for clustered MIMO channels with different array configurations. *IEEE Trans. Veh. Technol.* **2007**, *56*, 1924–1934. [[CrossRef](#)]
29. Rial, R.M.; Rusu, C.N.; Prelcic, G.; Alkhateeb, A.; Heath, R.W. Hybrid MIMO architectures for millimeter wave communications: Phase shifters or switches? *IEEE Access* **2016**, *4*, 247–267. [[CrossRef](#)]
30. Bajwa, W.U.; Haupt, J.; Sayeed, A.M.; Nowak, R. Compressed channel sensing: A new approach to estimating sparse multipath channels. *Proc. IEEE* **2010**, *98*, 1058–1076. [[CrossRef](#)]
31. Brady, J.; Behdad, N.; Sayeed, A.M. Beamspace MIMO for millimeter-wave communications: System architecture, modeling, analysis, and measurements. *IEEE Trans. Antennas Propag.* **2013**, *61*, 3814–3827. [[CrossRef](#)]
32. Candès, E.J.; Plan, Y. Matrix completion with noise. *Proc. IEEE* **2010**, *98*, 925–936. [[CrossRef](#)]
33. Weng, Z.; Wang, X. Low-rank matrix completion for array signal processing. In Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012; pp. 2697–2700.
34. Hale, E.T.; Yin, W.; Zhang, Y. Fixed-point continuation for  $\ell_1$ -minimization: Methodology and convergence. *SIAM J. Optim.* **2008**, *19*, 1107–1130. [[CrossRef](#)]
35. Larsen, R.M. Propack-Software for Large and Sparse svd Calculations. Available online: <http://sun.stanford.edu/~rmunk/PROPACK/> (accessed on 1 October 2019).
36. Chiang, K.Y.; Hsieh, C.; Dhillon, I.S. Matrix completion with noisy side information. *Adv. Neural Inf. Proc. Syst.* **2015**, *28*, 3447–3455.
37. Beck, A.; Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imag. Sci.* **2009**, *2*, 183–202. [[CrossRef](#)]
38. Park, S.; Park, J.; Yazdan, A.; Heath, R.W. Exploiting spatial channel covariance for hybrid precoding in massive MIMO systems. *IEEE Trans. Signal Process.* **2017**, *65*, 3818–3832. [[CrossRef](#)]
39. Groschel, P.; Zarei, S.; Carlowitz, C.; Lipka, M.; Sippel, E.; Ali, A.; Weigel, R.; Schober, R.; Vossiek, M. A system concept for online calibration of massive MIMO transceiver arrays for communication and localization. *IEEE Trans. Microw. Theory Technol.* **2017**, *65*, 1735–1750. [[CrossRef](#)]
40. Liu, L.; Hong, W.; Wang, H.; Yang, G.; Zhang, N.; Zhao, H.; Chang, J.; Yu, C.; Yu, X.; Tang, H.; et al. Characterization of Line-of-Sight MIMO Channel for Fixed Wireless Communications. *IEEE Antennas Wirel. Prop. Lett.* **2007**, *6*, 36–39. [[CrossRef](#)]
41. Xue, C.; He, S.; Ou, F.; Wei, M.; Huang, Y.; Yang, L. Asymmetric Subarray Structure Design for mmWave LoS MIMO Communication Systems. In Proceedings of the 2016 IEEE/CIC International Conference on Communications in China (ICCC), Chengdu, China, 27–29 July 2016.

