

Compressive Sensing Based Pilot Design For Sparse Channel Estimation in OFDM Systems

Roozbeh Mohammadian, Arash Amini, and Babak Hossein Khalaj

Abstract—We consider the deterministic pilot design problem for sparse channel estimation in an OFDM system. Our design is based on minimizing the coherence measure of the Fourier submatrix associated with the pilot subcarriers. This is done by optimizing over both pilot locations and pilot powers. As finding such global minimizer is a combinatorial problem, we resort to a greedy pilot allocation method. The resulting method achieves a suboptimal solution in a sequential manner and with reasonable computational complexity. Simulation results demonstrate that the proposed scheme performs similar to the existing methods with significantly lower computational complexity.

Index Terms—Compressive sensing, Deterministic pilot design, OFDM, Sparse channel estimation.

I. INTRODUCTION

RECENT studies reveal that incorporating channel sparsity in estimation of certain OFDM channels results in both higher estimation quality and lower pilot overhead [1]. Inline with the theories of compressed sensing [2], it is shown in [1] that uniformly at random pilot locations guarantee perfect channel recovery with very high probability.

To close the gap between theory and practice, the problem of deterministic pilot design has been considered in [3]–[6]. The design criterion in all these works is the coherence measure of the corresponding sensing matrix. In [3] and [4], it has been shown that pilot locations corresponding to cyclic difference sets (CDS) are optimal in terms of the coherence measure; however, such locations exist only for specific number of pilots and subcarriers. In settings where no CDS exists (most of the practical scenarios), [3] and [4] propose suboptimal and greedy methods. Almost difference sets (ADS) are the optimal choices when no CDS exists; [5] investigates how to simplify the combinatorial search that leads to ADSs. Besides the locations, pilot values are also optimized in [5]. Joint pilot locations and values (powers) are further studied in [6] by applying sequential stochastic search and second-order cone programming (SOCP).

In this letter, by adopting the coherence measure as the penalty function, we propose a new method for joint design of pilot locations and values, without assuming the existence of CDS or ADS. Our method is fully deterministic in contrast to the stochastic search of [6], and we avoid computationally intensive optimizations such as SOCP that are difficult to

implement in high dimensions. The result is a fast greedy technique that performs no worse than (and sometimes better than) the existing competitors with significantly lower computational complexity.

II. PILOT DESIGN FORMULATION

Let us consider an OFDM system with N subcarriers, among which N_P are used for pilot transmission. We denote the set of pilot subcarriers by $\mathcal{P}_{N_P} = \{p_1, p_2, \dots, p_{N_P}\}$, where $\mathcal{P}_{N_P} \subset \{1, \dots, N\}$. Expressing the pilot value transmitted at location p_i by $x(p_i)$ and defining $\mathbf{X} = \text{diag}\{x(p_1), \dots, x(p_{N_P})\}$, we can describe the vector of received signal at pilot subcarriers as

$$\mathbf{y} = \mathbf{X} \cdot \mathbf{F} \cdot \mathbf{h} + \mathbf{n}. \quad (1)$$

Here, $\mathbf{h} = [h(1), \dots, h(L)]^T$ is the channel impulse response of length $L \leq N$, \mathbf{F} is an $N_P \times L$ DFT submatrix where $\mathbf{F}(i, l)$ is defined as $e^{-j\frac{2\pi}{N}(p_i-1)(l-1)}$, and $\mathbf{n} = [n(p_1), n(p_2), \dots, n(p_{N_P})]^T$ stands for the vector of noise at pilot subcarriers. Note that (1) is valid for constant or slowly-varying channels that could be assumed constant within the time frame of an OFDM symbol. By defining $\Phi = \mathbf{X} \cdot \mathbf{F}$, we are able to rewrite (1) in the form

$$\mathbf{y} = \Phi \mathbf{h} + \mathbf{n}. \quad (2)$$

Thus, to estimate the channel we need to solve the linear inverse problem (2) for \mathbf{h} . Here, we have N_P observations and L unknowns. When effective delay spread of the channel exceeds the number of pilots, *i.e.*, $N_P < L$, (2) turns into an underdetermined problem.

Motivated by sparse structure of certain wireless channels, we assume \mathbf{h} is a k -sparse vector with $k < \frac{1}{5}L$; *i.e.*, \mathbf{h} contains at most k nonzero elements. Now, it might be possible to recover or approximate \mathbf{h} from the underdetermined system of (2) by finding the sparsest solution. To guarantee such a recovery, it is sufficient that the measurement matrix Φ has a small coherence value. The coherence of a generic matrix Φ denoted by μ_Φ is defined as [7]

$$\mu_\Phi = \max_{1 \leq i, l \leq L, i \neq l} \frac{|\langle \phi_i, \phi_l \rangle|}{\|\phi_i\|_2 \cdot \|\phi_l\|_2}, \quad (3)$$

where ϕ_i is the i -th column of Φ . It is shown that if $\mu_\Phi < \frac{1}{2k-1}$, then, any k -sparse vector \mathbf{h} can be uniquely recovered from $\Phi \mathbf{h}$ using a wide range of recovery techniques including the Orthogonal Matching Pursuit (OMP) [8]. For the matrix Φ in our OFDM channel estimation, we can write (3) as

$$\mu_\Phi = \frac{\max_{r \in \mathcal{L}} \left| \sum_{i=1}^{N_P} |x(p_i)|^2 e^{-j\frac{2\pi}{N} p_i r} \right|}{\sum_{i=1}^{N_P} |x(p_i)|^2}, \quad (4)$$

The authors are with the Department of Electrical Engineering and Advanced Communication Research Institute (ACRI), Sharif University of Technology, Tehran, Iran. B. H. Khalaj is also with the School of Computer Science, Institute for Research in Fundamental Sciences (IPM), Tehran 19538-33511, Iran (e-mails: rmohammadian@ee.sharif.ir, {aamini, khalaj}@sharif.ir).

Manuscript received April 16, 2016; revised xxx xx, 2016.

where $\mathcal{L} = \{1, \dots, L-1\}$. Our goal in this paper is to minimize μ_{Φ} by selecting the pilot locations \mathcal{P}_{N_P} and pilot values $\{x(p_i)\}_{i=1}^{N_P}$. We observe in (4) that the contribution of pilot values in the coherence measure is through their magnitudes $v(i) = |x(p_i)|^2$; hence, we simplify the problem of pilot design as

$$\Omega_{\text{opt}} = \underset{\Omega}{\operatorname{argmin}} \max_{r \in \mathcal{L}} \left| \sum_{i=1}^{N_P} v(i) e^{-j \frac{2\pi}{N} p_i r} \right|. \quad (5)$$

Here, $\Omega = (\mathcal{P}_{N_P}, \mathbf{v})$ where $\mathbf{v} = [v(1), \dots, v(N_P)]^T$ with the additional constraint that $\sum_{i=1}^{N_P} v(i) = \mathbf{1}^T \mathbf{v} = 1$. Note that the latter constraint does not tarnish the generality of the design, as by scaling \mathbf{v} the coherence measure remains unchanged.

The pilot design problem of (5) is a combinatorial optimization, and it is intractable to find its exact solution. Although the pilot design is mainly a preprocessing block and is implemented only once, its computational complexity becomes significant when N and N_P become very large, or in particular applications where we regularly need to redesign the pilot pattern. One such example is the OFDM-based cognitive radio (CR) systems for which the available spectrum, and therefore subcarriers, are changing over time. In such cases, a suboptimal design with reasonable computational cost is preferable [6]. In the sequel, we propose a low-cost suboptimal solution that divides (5) into separate problems for setting \mathcal{P}_{N_P} and \mathbf{v} , and sequentially iterates between them.

III. PILOT POWER ALLOCATION

We first study how to assign pilot powers to a given set of pilot locations $\mathcal{P}_m = \{p_1, \dots, p_m\}$. Indeed, in our final design, we start from $m = 1$ and gradually move towards $m = N_P$. By defining

$$a(r) = \sum_{i=1}^m v(i) e^{-j \frac{2\pi}{N} p_i r}, \quad r \in \mathcal{L}, \quad (6)$$

the problem of optimal power assignment translates into

$$\mathbf{v}_{\text{opt}} = \underset{\mathbf{v} \geq 0, \mathbf{1}^T \mathbf{v} = 1}{\operatorname{argmin}} \max_{r \in \mathcal{L}} |a(r)| = \underset{\mathbf{v} \geq 0, \mathbf{1}^T \mathbf{v} = 1}{\operatorname{argmin}} \|\mathbf{a}\|_{\infty}, \quad (7)$$

where \mathbf{a} stands for the $(L-1) \times 1$ vector of $a(r)$ values and

$$\|\mathbf{a}\|_{\infty} = \lim_{q \rightarrow \infty} \|\mathbf{a}\|_q = \lim_{q \rightarrow \infty} \left[\sum_{r=1}^{L-1} |a(r)|^q \right]^{\frac{1}{q}} \quad (8)$$

is the ℓ_{∞} -norm of \mathbf{a} . As indicated in [6], (7) is a SOCP problem and can be solved using available optimization packages such as MOSEK [9]. However, the involved interior point methods in such solvers, make them very slow in moderate to high dimensions [10]. In this paper, we slightly approximate the cost in (7) and apply a fast gradient method instead. This results in a first-order method that is computationally easy to implement.

Since $\|\mathbf{a}\|_{\infty}$ is not differentiable, in (7) we use its smooth approximation $\|\mathbf{a}\|_q$, where $q > 1$ is set large enough (we use $q = 20$ in our simulations). Further, we replace the constraints $\mathbf{1}^T \mathbf{v} = 1$ and $\mathbf{v} \geq 0$ in (7) with $v(i) \geq v_{\min}$ for all $i = 1, \dots, m$, where $v_{\min} > 0$ is a given parameter. Although the two constraint sets are not equivalent for the purpose of minimization, with the latter we obtain a set of

TABLE I
POWER ALLOCATION ALGORITHM (ALGORITHM I).

Input: $\alpha_0 > 0, \epsilon > 0, v_{\min}, I_{\max}$.
1: Initialization: $n \leftarrow 1, \mathbf{v}^{(0)} \leftarrow \frac{1}{L} \mathbf{1}, \mathbf{x}^{(0)} \leftarrow \mathbf{v}^{(0)}, \mathbf{v}^{(1)} \leftarrow \mathbf{0}, \theta_1 \leftarrow 1$.
2: while $(f_{\mathcal{P}_m}(\mathbf{v}^{(n)}) - f_{\mathcal{P}_m}(\mathbf{v}^{(n-1)}) > \epsilon \text{ and } n < I_{\max})$
3: Set $\mathbf{u} \leftarrow (1 - \theta_n) \mathbf{v}^{(n-1)} + \theta_n \mathbf{x}^{(n-1)}$.
4: Compute the normalized gradient $\mathbf{d} \leftarrow \nabla f_{\mathcal{P}_m}(\mathbf{u}) / \ \nabla f_{\mathcal{P}_m}(\mathbf{u})\ _2$.
5: Set $\mathbf{u} \leftarrow \mathbf{u} - \alpha_0 \mathbf{d}$.
6: for $i = 1 : m$
7: if $u(i) < v_{\min}$
8: $u(i) \leftarrow v_{\min}$
9: end if
10: end for (i)
11: Set $\mathbf{v}^{(n)} \leftarrow \mathbf{u}$
12: Set $\mathbf{x}^{(n)} \leftarrow \mathbf{v}^{(n-1)} + \frac{1}{\theta_n} (\mathbf{v}^{(n)} - \mathbf{v}^{(n-1)})$.
13: $n \leftarrow n + 1$.
14: $\theta_n \leftarrow 2/(n + 1)$.
15: end while
16: $\mathbf{v}_{\text{opt}} \leftarrow \mathbf{v}^{(n)} / \ \mathbf{v}^{(n)}\ _1$.

pilot powers with less variance which is desirable in practice. These modifications result in

$$\mathbf{v}_{\text{opt}} = \underset{\mathbf{v} \geq v_{\min}}{\operatorname{argmin}} f_{\mathcal{P}_m}(\mathbf{v}), \quad (9)$$

where

$$f_{\mathcal{P}_m}(\mathbf{v}) = \sum_{r \in \mathcal{L}} \left| \sum_{i=1}^m v(i) e^{-j \frac{2\pi}{N} p_i r} \right|^q. \quad (10)$$

Note that $f_{\mathcal{P}_m}$ is both convex and differentiable for $q > 1$. Moreover, its gradient $\nabla f_{\mathcal{P}_m}(\mathbf{v})$ is given by

$$q \sum_{r=1}^{L-1} |a(r)|^{q-2} \operatorname{Re} \left\{ a(r) [e^{j \frac{2\pi}{N} p_1 r}, \dots, e^{j \frac{2\pi}{N} p_m r}]^H \right\}. \quad (11)$$

Therefore, it is possible to apply the Nesterov's accelerated gradient method [11] to minimize (9). The details of the implementation are presented in Table I. The parameters $\alpha_0 > 0$, ϵ , v_{\min} , and I_{\max} are the inputs of Algorithm I, representing the step size, the stopping measure, the minimum acceptable pilot power and the maximum allowed number of iterations, respectively. After the initialization step, Algorithm I estimates a candidate \mathbf{u} for the minimizer of the cost in each iteration. This estimate is obtained based on the previous estimates and the current gradient of the cost (steps 3 – 5). Then, a thresholding block (steps 6 – 10) enforces the constraint $v(i) \geq v_{\min}$. Indeed, by thresholding we project \mathbf{u} onto the feasible set. This way, we obtain $\mathbf{v}^{(n)}$ which is the estimate of the minimizer \mathbf{v}_{opt} at iteration n .

A. Computational complexity

In each iteration of Algorithm I we have $(L-1)(9m+1) + 4m + 1$ summations, $(L-1)(11m+5/2q+2) + 6m+4$ multiplications, and $2(L-1) + 1$ square-roots. By taking the maximum number of iterations I_{\max} into account, we observe that in total, there are around $20mL I_{\max}$ operations performed in Algorithm I.

TABLE II
JOINT PILOT DESIGN ALGORITHM (ALGORITHM II).

Input: $N, N_P, L, I_{\text{out}}$.
1: Initialization: Set $\mathcal{P}_1^* \leftarrow \{1\}$.
2: for $n = 2, \dots, N_P$
3: Obtain \mathcal{P}_n^* according to (13).
4: end for (n)
5: Set $\hat{\mathcal{P}} \leftarrow \mathcal{P}_{N_P}^*, \mathcal{P}_{\text{opt}} \leftarrow \mathcal{P}_{N_P}^*$ and $J_{\text{opt}} \leftarrow J(\mathcal{P}_{N_P}^*)$
6: for $i = 1, \dots, I_{\text{out}}$
7: for $n = 1, \dots, N_P$
8: Obtain p^* according to (15).
9: if $J(\hat{\mathcal{P}}_{p^* \setminus n}) < J(\hat{\mathcal{P}})$
10: Update $\hat{\mathcal{P}}$ by replacing p_n^* with p^*
11: end if
12: end for (n)
13: if $\mathcal{P}_{\text{opt}} = \hat{\mathcal{P}}$
14: break.
15: end if
16: Set $\mathcal{P}_{\text{opt}} \leftarrow \hat{\mathcal{P}}$.
17: end for (i)
18: Find \mathbf{v}_{opt} corresponding to \mathcal{P}_{opt} , using Algorithm I.

IV. JOINT PILOT PATTERN AND POWER DESIGN

In this section we propose a joint pilot pattern and power design method for minimizing μ_{Φ} . The proposed algorithm consists of two main parts. In the first part, N_P pilot locations are determined sequentially using a greedy approach. In the second part, we try to improve the achieved pattern by single replacement of the pilot locations. The latter task is repeated for a maximum of I_{out} iterations, where I_{out} is considered as one of the input parameters. The detailed algorithm is presented in Table II.

A. First Part

Since μ_{Φ} remains unchanged by circularly shifting the pilot locations, without loss of generality, we initialize the set of pilot locations with $\mathcal{P}_1^* = \{1\}$. Next, we increase the size of this set up to N_P using the following procedure: let $\mathcal{P}_{n-1}^* = \{p_1^*, \dots, p_{n-1}^*\}$ be the current set of pilot locations with $n-1$ elements. To add the n th location, we consider all possible choices $p \in \{\mathcal{N} \setminus \mathcal{P}_{n-1}^*\}$ and form $\mathcal{P}_{n|p} = \mathcal{P}_{n-1}^* \cup p$. We set the corresponding pilot powers by minimizing

$$J(\mathcal{P}_{n|p}) = \min_{\mathbf{v} \geq \mathbf{v}_{\min}} f_{\mathcal{P}_{n|p}}(\mathbf{v}). \quad (12)$$

Finally, we find \mathcal{P}_n^* as the overall best solution

$$\mathcal{P}_n^* = \underset{\mathcal{P}_{n|p}}{\operatorname{argmin}} J(\mathcal{P}_{n|p}). \quad (13)$$

B. Second Part

In the previous part, we followed a greedy approach to determine the desired pilot pattern of size N_P . In this second part, we try to improve μ_{Φ} by updating the pattern. More precisely, we discard one of the pilot locations iteratively and find the best substitution for it. With this technique, we shall finally obtain a pilot pattern which is locally optimal in the sense that no single substitution improves its coherence.

We implement this procedure using two nested loops called outer and inner iterations. The outer iterations control the number of replacements, and continue until either a maximum

TABLE III
PILOT PATTERNS DESIGNED BY DIFFERENT METHODS.

Method	N_P	μ_{Φ}	Run-time	\mathcal{P}_{N_P}
Proposed	16	0.2582	18.47 [s]	1, 21, 53, 69, 97, 125, 133, 141, 189, 201, 213, 217 225, 237, 249, 253
Method of [6]	16	0.2487	88680 [s]	1, 25, 61, 65, 109, 125, 141, 149, 153, 181, 205, 209 229, 237, 245, 249
Proposed	20	0.2130	16.19 [s]	1, 17, 21, 29, 33, 45, 49, 53, 57, 97, 125, 133 141, 149, 181, 189, 201, 213, 241, 253
Method of [6]	20	0.2177	159780 [s]	11, 15, 18, 31, 35, 43, 50, 60, 86, 94, 119, 123 134, 155, 162, 208, 216, 220, 228, 250
Proposed	25	0.1787	13.71 [s]	1, 13, 17, 21, 29, 33, 49, 53, 57, 73, 77, 121, 125, 133 141, 173, 177, 181, 189, 201, 213, 217, 225, 241, 253
Method of [6]	25	0.1810	252300 [s]	3, 13, 17, 25, 29, 33, 41, 47, 54, 71, 92, 99, 117, 132 135, 152, 159, 164, 184, 201, 205, 209, 213, 247, 256

number of iterations is achieved or no further replacement improves the coherence. In the inner loop, we consider the N_P locations separately and check whether they could be replaced. Let $\hat{\mathcal{P}} = \{p_1^*, \dots, p_{N_P}^*\}$ be the available set of pilot locations at the beginning of the n -th inner iteration ($1 \leq n \leq N_P$), and let $J(\hat{\mathcal{P}})$ be the corresponding cost. For each $p \in \mathcal{N} \setminus \hat{\mathcal{P}}$ we define $\hat{\mathcal{P}}_{p \setminus n} = \{\hat{\mathcal{P}} \setminus p_n^*\} \cup \{p\}$ and evaluate

$$J(\hat{\mathcal{P}}_{p \setminus n}) = \min_{\mathbf{v} \geq \mathbf{v}_{\min}} f_{\hat{\mathcal{P}}_{p \setminus n}}(\mathbf{v}). \quad (14)$$

Next, we find the best substitute for p_n^* by

$$p^* = \underset{p \in \mathcal{N} \setminus \hat{\mathcal{P}}}{\operatorname{argmin}} J(\hat{\mathcal{P}}_{p \setminus n}). \quad (15)$$

If $J(\hat{\mathcal{P}}_{p^* \setminus n})$ is less than $J(\hat{\mathcal{P}})$, we update $\hat{\mathcal{P}}$ by replacing p_n^* with p^* ; otherwise, $\hat{\mathcal{P}}$ remains unchanged.

C. Computational complexity

Based on the derived computational cost of Algorithm I in Section III-A, the first and second parts of Algorithm II include around $10LI_{\max}N_P^2N$ and $20LI_{\max}I_{\text{out}}N_P^2N$ operations, respectively, where I_{out} stands for the maximum number of outer iterations. Computationally, the first part is counted half a single outer iteration of the second part. Hence, the overall cost is practically determined by the second part.

The computational complexity of the method in [6], as indicated in [6], is at least

$$\mathcal{O}(T_1T_2(L-1)^{1.5}(N_P+1)^3N_P(N-N_P+1)), \quad (16)$$

where T_1 and T_2 are, respectively, the number of outer and inner iterations.

V. SIMULATION RESULTS

In this section, we present the result of a number of numerical simulations. In all experiments, the input parameters are set as $\alpha_0 = 6.4 \times 10^{-3}$, $\epsilon = 10^{-16}$, $q = 20$, $v_{\min} = 10^{-3}$, $I_{\max} = 20$, and $I_{\text{out}} = 10$. Also, the considered OFDM system contains $N = 256$ subcarriers and a cyclic prefix of length $L = 60$. For the recovery of sparse channels, we employ the OMP algorithm [8] and evaluate the average performance of the reported methods over 5000 trials in each setting.

In our first experiment, we compare the performance of Algorithm II based on Algorithm I with the method of [6] that employs SOCP. For the method of [6], we set the parameters as recommended by the authors; in particular, T_1 and T_2 are set as

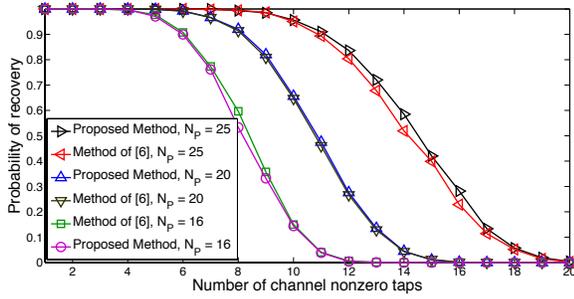


Fig. 1. Recovery probability for different pilot design schemes.

1000 and 15, respectively. The results including the coherence measure, computation time and the obtained pilot locations are presented in Table III. The experiment is conducted for $N_P = 16$, $N_P = 20$ and $N_P = 25$. The reported experiments in Table III indicate that our proposed method results in about 99% reduction in run time compared to that of the method in [6] under the considered setup, while the coherence measures are similar.

In the second experiment, we study the probability of exact channel recovery in a noiseless setting where the pilots are set as obtained in the previous experiment (Table III); here, the exact recovery corresponds to the reconstruction SNR of at least 100 dB. Figure 1 shows that Algorithm II and method of [6] perform similarly at various sparsity levels with slight advantage for Algorithm II.

Next, we investigate the performance of the channel estimation in noisy settings. For this purpose, we present both the channel estimation mean-square error (MSE) and the overall bit error rate (BER) of the OFDM system (QPSK modulation) in Figures 2 and 3, respectively. For this experiment, we consider two scenarios: 1) channel sparsity of order $k = 5$ with $N_P = 16$ pilots, and 2) channel sparsity of order $k = 6$ with $N_P = 20$ pilots. In addition to Algorithm II and the method of [6], we also include the random pilot allocation in our comparisons. For the latter, we select pilot locations uniformly at random and set their powers equally. Moreover, random pilots are updated in each realization.

To simulate the sparse multipath channel, we determine the non-zero taps uniformly at random within the available window of length L . The tap values are set by realizations of i.i.d. complex-valued Gaussian random variables with zero-mean and unit variance. After applying the multipath channel on the data, we include the additive complex-valued Gaussian white noise; the variance of the noise is determined such that a desired level of overall SNR is achieved.

To provide a reference for the estimation of the sparse channels, we have included the oracle estimator in Figures 2 and 3. The oracle estimator knows the location of non-zero taps beforehand and uses pilots only to extract the tap values (least square estimation).

Figures 2 and 3 both reconfirm that Algorithm II and the method of [6] perform similarly, and outperform the random pilot design.

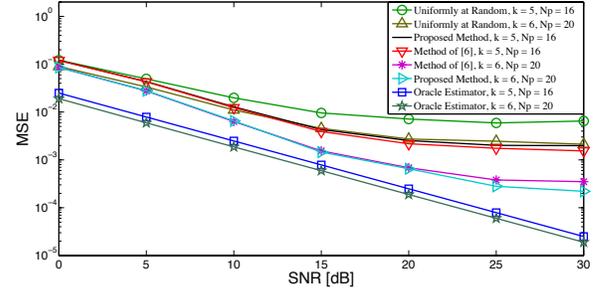


Fig. 2. MSE of channel estimation for different schemes.

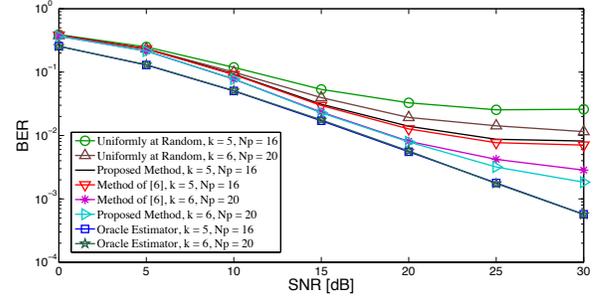


Fig. 3. BER performance of different pilot design schemes.

VI. CONCLUSION

A new deterministic pilot design scheme for sparse channel estimation in OFDM systems is proposed. The method is based on minimizing the coherence of the Fourier submatrix associated with the pilot subcarriers. The formulation of the optimization problem jointly determines the pilot locations and the pilot powers. Simulation results demonstrate that the proposed method performs similar to the existing methods with significantly lower computational complexity.

REFERENCES

- [1] G. Taubock and F. Hlawatsch, "A compressed sensing technique for OFDM channel estimation in mobile environments: Exploiting channel sparsity for reducing pilots," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, April 2008, pp. 2885–2888.
- [2] E. J. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [3] P. Pakrooh, A. Amini, and F. Marvasti, "OFDM pilot allocation for sparse channel estimation," in *EURASIP J. Adv. Signal Process.*, vol. 59, 2012.
- [4] C. Qi, G. Yue, L. Wu, Y. Huang, and A. Nallanathan, "Pilot design schemes for sparse channel estimation in OFDM systems," *IEEE Trans. Veh. Tech.*, vol. 64, no. 4, pp. 1493–1505, April 2015.
- [5] M. Khosravi and S. Mashhadi, "Joint pilot power and pattern design for compressive OFDM channel estimation," *IEEE Comm. Lett.*, vol. 19, no. 1, pp. 50–53, January 2015.
- [6] C. Qi, L. Wu, Y. Huang, and A. Nallanathan, "Joint design of pilot power and pilot pattern for sparse cognitive radio systems," *IEEE Trans. Veh. Tech.*, vol. 64, no. 11, pp. 5348–5390, November 2015.
- [7] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. Inform. Theory*, vol. 47, no. 7, pp. 2845–2862, November 2001.
- [8] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Trans. Inform. Theory*, vol. 50, no. 10, pp. 2231–2242, October 2004.

- [9] MOSEK, *The MOSEK optimization toolbox for MATLAB manual Version 7.1 (Revision 28)*, 2015. [Online]. Available: <http://docs.mosek.com/7.1/toolbox/index.html>
- [10] X. Chen, Q. Lin, S. Kim, J. G. Carbonell, and E. P. Xing, “Smoothing proximal gradient method for general structured sparse regression,” *Ann. Appl. Stat.*, vol. 6, no. 2, pp. 719–752, June 2012.
- [11] Y. Nesterov, *Introductory Lectures on Convex Optimization. A Basic Course*, 1st ed. Springer US, 2004.