Signal Processing 167 (2020) 107318

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Parameter estimation and channel reconstruction based on compressive sensing for ultra-wideband MB-OFDM systems

Taoyong Li[a,b,*], Brecht Hanssens[c], Wout Joseph[c], Heidi Steendam[a]

[a] Telecommunications and Information Processing Department Ghent University\IMEC, Gent, Belgium
[b] Collaborative Innovation Center of Information Sensing and Understanding, Xi'an, China
[c] WAVES Research Group, Ghent University\IMEC, Belgium

ARTICLE INFO

Article history:
Received 22 May 2019
Revised 11 September 2019
Accepted 24 September 2019
Available online 24 September 2019

Keywords:
Multi-band orthogonal frequency-division multiplexing (MB-OFDM)
Ultra-wideband (UWB)
Parameter estimation
Compressive sensing
Channel impulse response

ABSTRACT

Multi-band orthogonal frequency-division multiplexing (MB-OFDM) is an important transmission technique for ultra-wideband (UWB) communication. One of the challenges for practical realization of these UWB MB-OFDM systems is the estimation of the channel. In UWB MB-OFDM, the channel can be modelled as sparse, and channel estimation (CE) based on compressed sensing (CS) can be used. However, the existing techniques all require prior knowledge of some channel parameters, which are not known in practice, e.g. the dictionary size, corresponding to the effective duration of the channel impulse response (CIR), and the sparsity of the CIR. Therefore, in this paper, we propose a CS-based channel parameter estimation method to estimate the dictionary size and the sparsity based on a pilot preamble of which the duration is shorter than the total duration of the CIR. Using the resulting parameter estimates, we reconstruct the CIR with the compressive sampling matching pursuit (CoSaMP) method. We show that the proposed algorithm is able to accurately estimate the sparsity and the dictionary size, and can effectively reconstruct the CIR for channels that are either based on a mathematical model or real, measured channels. Moreover, as the algorithm has acceptable complexity, the proposed method is suitable for practical use.

1. Introduction

Multi-band orthogonal frequency-division multiplexing (MB-OFDM) is a technique that is considered as one of the most promising techniques for ultra wideband (UWB) transmission, thanks to its ability to mitigate the effects of multipath fading and interference, and to achieve a high spectral efficiency at a relatively low cost [1,2]. One of the issues that needs to be solved in practical UWB MB-OFDM systems comprises the estimation of the channel. To meet this challenge, UWB MB-OFDM adopts a frame-based transmission [3], where pilot sequences are included in the frame preamble for channel estimation (CE). However, as the channel impulse response (CIR) in UWB MB-OFDM is very long, long pilot preambles must be used to accurately estimate the channel. As long pilot preambles limit the data throughput, often the pilot preamble is shortened. Because of this shorter preamble, traditional channel estimators, such as least-squares (LS), maximum-likelihood (ML) and minimum mean-squared error (MMSE) estimators [4–7], fail to accurately estimate the CIR.

In indoor environments, typically the propagation environment is complex, and results in many reflections. At the same time, the resolution of the ultrawideband signal is very high, implying the system can identify many of the multipath components. As a consequence, the channel impulse response will typically be very long. However, measurements of the UWB indoor channel show that the multipath components are strongly clustered, implying the channel impulse response, although being widely dispersed in time, only contains a limited number of non-zero contributions, i.e. the channel can be modelled as sparse. For example, [8,9] demonstrate that indoor channel models considered for the IEEE 802.15.4a standard [10] are sparse. Moreover, this sparsity is shown to be enlarged when the signal resolution increases [9]. Also the different channel models for the IEEE 802.15.3a standard [11] can be described with a limited number of non-zero channel taps. Besides the theoretical channel models considered in the literature, we also performed a measurement campaign in a laboratory environment, and show in this paper that the resulting channel is sparse. Consequently, we can use compressive sensing (CS) methods [12,13] to reconstruct the CIR and achieve channel estimation. Recently, several CE

* Corresponding author at: Telecommunications and Information Processing Department Ghent University\IMEC, Gent, Belgium.
E-mail address: Taoyong.Li@ugent.be (T. Li).

https://doi.org/10.1016/j.sigpro.2019.107318
0165-1684/

algorithms based on CS for UWB communication have been developed [8,14–16]. In [8], the applicability of CS for UWB channel estimation is investigated, and the authors employ standard Matching Pursuit (MP) algorithms for CS, such as subspace pursuit (SP) [17], orthogonal matching pursuit (OMP) [18] and compressive sampling matching pursuit (CoSaMP) [19], to accurately reconstruct the CIR. In [14], the authors propose four practical dictionaries to increase the sparsity of UWB signals, so that the UWB signals can be reconstructed more efficiently. In [15], another CS technique, i.e. the Bayesian CS (BCS) algorithm [20], is employed to reconstruct UWB signals and obtain CE. Although the BCS algorithm achieves a better performance than the MP algorithms, it requires intensive computations, raising a barrier for practical implementation. In [16], a CS dictionary, called eigen-dictionary, is proposed, exploiting the statistical sparsity of UWB signals where the channel structure exhibits several clusters of significant channel coefficients. Based on this structure, two novel BCS algorithms are proposed to efficiently reconstruct UWB CIR. Common to all these CS-based CE algorithms is that they require prior knowledge of the parameters of the underlying CIR model, which is not available in practice. Without the knowledge of these parameters, the CIR can not be estimated accurately, deteriorating the performance of UWB MB-OFDM systems.

In this paper, we extend the CoSaMP algorithm from [19], which combines low complexity and good CE performance, to autonomously estimate the required channel parameters. The CoSaMP algorithm requires the knowledge about the dictionary size, of which the optimal value is strongly correlated to the effective CIR duration, i.e. the duration of the part of the CIR containing the dominant channel components, and the sparsity of the channel, i.e. the number of non-zero channel taps. Optimally, the dictionary size and sparsity must be estimated jointly. Several classical algorithms exist to achieve this joint estimation, e.g. the simplex algorithm [21]. However, the computational burden of these algorithms is very high, and therefore limit the applicability of these joint estimators. Therefore, as main novelty, we propose in this paper an algorithm that has low complexity compared to the abovementioned joint estimation algorithm. To this end, we first show that, although the dictionary size and sparsity are correlated, the optimal value of the dictionary size becomes essentially independent of the sparsity if the sparsity is sufficiently large. Based on this observation, we propose a two-step approach, where in the first phase, the optimal dictionary size is estimated, while in the second phase, the optimal value of the sparsity is obtained. In both phases, the algorithm adaptively searches for the optimal value of the parameter, using the pilot sequence included in the preamble. We show that the proposed adaptive CS-based parameter estimation algorithm not only can be applied to channels simulated based on a mathematical model, but also is able to exactly reconstruct the CIR measured in realistic scenarios. Although the proposed algorithm is sub-optimal in the sense that the mean-squared error of the resulting channel estimation is slightly higher than for the case where the simplex method is used, the resulting complexity is much smaller than with the simplex method, e.g. for short pilot preambles, the complexity of the proposed algorithm is 10 times lower than with the simplex method, and the difference in complexity increases when the length of the preamble increases. Further, we compare the performance of the proposed algorithm with state-of-the-art algorithms, and demonstrate that the proposed algorithm performs well, even if the pilot preamble is considerably shortened.

The rest of the paper is organized as follows. In Section 2, we introduce the channel model used for MB-OFDM systems and describe the measurement setup used to obtain the sparse measured channel. In Section 3, we briefly explain how CS is applied to the estimation of sparse channels, and we discuss the influence of the dictionary size and sparsity on channel reconstruction. The algorithm to estimate the dictionary size and the sparsity is introduced in Section 4. Further, we evaluate the complexity of the proposed algorithm in this section. In Section 5, we evaluate the performance of the proposed algorithm and compare its performance with that of state-of-the-art algorithms. Finally, the conclusions are given in Section 6.

2. Sparse channel

In this section, first we briefly introduce the CM for the IEEE 802.15.3a standard [11], suitable for UWB MB-OFDM systems that was used to generate the simulated channels, and then we describe the measurement setup that was used to obtain sparse measured channels to test our algorithm in realistic scenarios.

2.1. Channel model

The channel impulse response considered for the IEEE 802.15.3a standard [11] consists of a tapped-delay line model containing L clusters of K multipath components:

$$h(t) = X \sum_{l=1}^{L} \sum_{k=1}^{K} \alpha_{k,l} \delta(t - T_l - \tau_{k,l}), \tag{1}$$

where $\alpha_{k,l}$ are the multipath gain coefficients, T_l is the delay of the lth cluster, $\tau_{k,l}$ is the delay of the kth multipath component relative to the lth cluster arrival time T_l and the prefactor X corresponds to the log-normal shadowing. The delays T_l and $\tau_{k,l}$ follow an exponential distribution with cluster arrival rate Λ and ray arrival rate λ, respectively:

$$P(T_l|T_{l-1}) = \Lambda \exp[-\Lambda(T_l - T_{l-1})] \tag{2}$$

$$P(\tau_{k,l}|\tau_{k-1,l}) = \lambda \exp[-\lambda(\tau_{k,l} - \tau_{k-1,l})]. \tag{3}$$

We select $\tau_{0,l} = 0$. The multipath gain coefficient $\alpha_{k,l}$ in (1) can be decomposed as follows:

$$\alpha_{k,l} = p_{k,l} \zeta_l \beta_{k,l}, \tag{4}$$

where $p_{k,l}$ equiprobably takes the values ± 1 to account for signal inversions due to reflections, ζ_l represents the fading associated with the lth cluster, and $\beta_{k,l}$ corresponds to the fading associated with the kth ray of the lth cluster. This fading coefficient $\zeta_l \beta_{k,l}$ follows a log-normal distribution:

$$20 \log_{10}(\zeta_l \beta_{k,l}) \sim N(\mu_{k,l}, \sigma_1^2 + \sigma_2^2), \tag{5}$$

where σ_1 is the standard deviation from the cluster log-normal fading term ζ_l and σ_2 is the standard deviation from the ray log-normal fading term $\beta_{k,l}$. Further, defining the cluster decay factor Γ and ray decay factor γ, the mean $\mu_{k,l}$ can be written as:

$$\mu_{k,l} = \frac{10 \ln(\Omega_0 - 10T_l/\Gamma - 10\tau_{k,l}/\gamma)}{\ln(10)} - \frac{(\sigma_1^2 + \sigma_2^2)\ln(10)}{20}, \tag{6}$$

where Ω_0 is the average energy of the first path of the first cluster. Finally, the log-normal shadowing factor X of the total multipath power from (1) has the distribution:

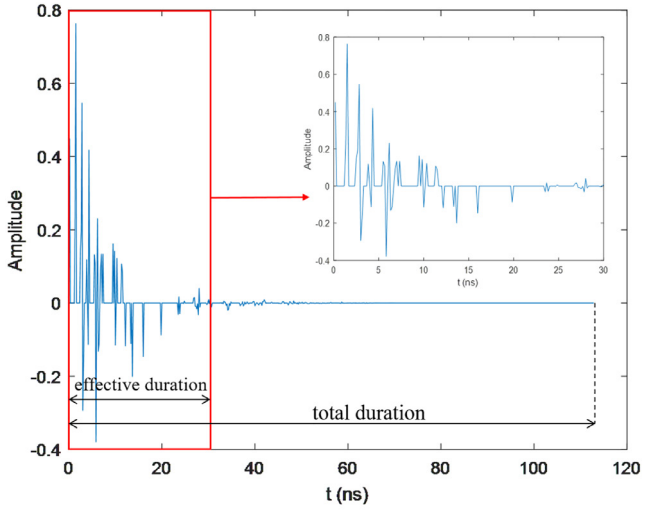
$$20 \log_{10}(X) \sim N(0, \sigma_x^2), \tag{7}$$

where σ_x is the standard deviation of the log-normal shadowing of the total multipath power.

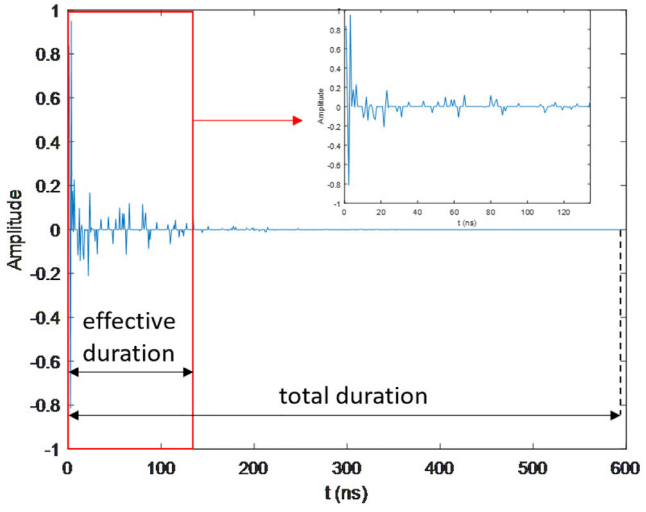
The parameters of the four channel models presented in [11] are listed in Table 1. These four models consider communication among UWB devices located within a range of less than 10 m. Specifically, CM1 and CM2 model the line-of-sight (LOS) and non-LOS (NLOS) channel environments, for ranges smaller than 4 m. For larger ranges, the NLOS models CM3 and CM4 are used, with emphasis on the strong delay dispersion Γ from CM4 [2]. In this paper, we consider discrete-time channel models derived from the

Table 1
Parameters of the Four Channel Models (CMs) from [11].

Parameters	CM1	CM2	CM3	CM4
Λ	0.0233	0.4	0.0667	0.0667
λ	2.5	0.5	2.1	2.1
Γ	7.1	5.5	14.00	24.00
γ	4.3	6.7	7.9	12
σ_1 (dB)	3.3941	3.3941	3.3941	3.3941
σ_2 (dB)	3.3941	3.3941	3.3941	3.3941
σ_x (dB)	3	3	3	3



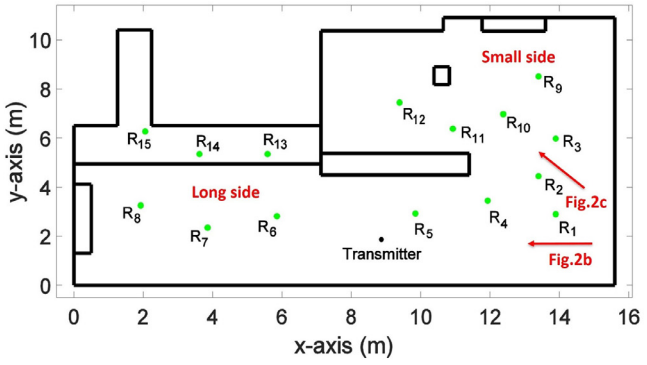
(a)



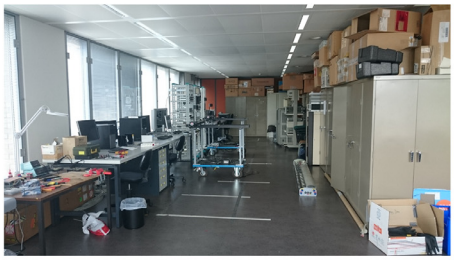
(b)

Fig. 1. Sparse channel: (a) an example channel realization of CM1 with tap spacing of $t = 1/6$ ns. (b) measured channel between transmitter and R_1 receiver with tap spacing of $t = 4/5$ ns.

above continuous-time models. Following the IEEE 802.15.3a standard [11], the discrete-time CIR $\mathbf{h} = [h(0), h(1), \ldots, h(L_{taps} - 1)]$ can be obtained by oversampling the continuous-time CIR $h(t)$, followed by an anti-aliasing filtering, down conversion and decimation. As an example, we show in Fig. 1(a) the discrete-time domain CIR for a realization of CM1 with tap spacing $t = 1/6$ ns, together with a close-up of the 180 first taps. As can be observed, the number of dominant taps, having a non-negligible amplitude, is quite



(a)



(b)

(c)

Fig. 2. Measurement environment: (a) Schematic representation, (b) Long side of laboratory, (c) Small side of laboratory.

small, i.e. most taps have a (close to) zero amplitude. Further, the dominant taps are confined in the first part of the CIR, i.e. the tail contains only close-to-zero taps. Hence, the effective duration of the CIR, corresponding to this dominant, first part of the CIR, is much smaller than the total duration of the CIR. Similar results are obtained with the other three channel models. Therefore, the channel can be considered as sparse in the time domain, indicating that compressive sensing methods to reconstruct the CIR can be employed.

2.2. Measured channel

The experiments to measure the sparse channels were carried out in a laboratory of Ghent University in Belgium. The laboratory, shown in Fig. 2, roughly has an L-shaped form. The long side (Fig. 2(b)) approximately has length 16 m and width 5 m, while the small side (Fig. 2(c)) approximately has length 8.5 m and width 5 m. We selected one transmitter position and 15 receiver positions (see Fig. 2(a)). Of these positions, 8 positions (i.e. $R_1 - R_8$) were considered as light-of-sight positions, where a free space path exists between transmitter and receiver, 4 positions (i.e. $R_9 - R_{12}$) were regarded as obstructed-LoS (OLoS), where the signals undergo a reflection and/or a diffraction, and 3 positions (i.e. $R_{13} - R_{15}$) are non-LoS (NLoS) scenarios, as a plasterboard wall is present between transmitter and receiver.

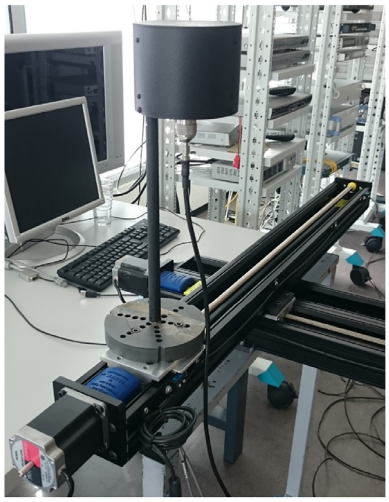


Fig. 3. UWB antenna with an automated positioning system.

At both the transmitter and receiver sides, omnidirectional UWB antennas of type electrometrics EM-6865 [22] were placed 1.5 m above ground level, as shown in Fig. 3. To measure the complex gain for each combination of transmit and receive pair, we used a Rohde & Schwarz ZNB8 vector network analyzer (VNA). The VNA calibration included the feeder cables to eliminate their effect on the measurement data. We created at both sides a [4 × 4] virtual antenna array by using an automated positioning systems. At each of the receiver positions, channel measurements were performed, and the channel was obtained by averaging these measurement data. An example of the resulting discrete-time CIR for receiver position R_6 is shown in Fig. 1(b), for a tap spacing $t = 4/5$ ns. As can be observed, the CIR contains only a few dominant taps, i.e. most taps have a near-zero or zero amplitude, implying the channel can indeed be considered as sparse. Similar results were obtained for the other receiver positions.

3. Compressive sensing based channel reconstruction scheme

In this section, we discuss the reconstruction of the CIR using CS methods, and analyze the effect of two parameters, i.e. the dictionary size and sparsity of CIR, on the CE. Let us first revisit the principle of CS. Let us assume we want to reconstruct a sparse signal $\boldsymbol{\varphi}$ from an observation $\boldsymbol{\rho}$

$$\boldsymbol{\rho} = \boldsymbol{\Upsilon}\boldsymbol{\varphi} + \mathbf{n}_0 \tag{8}$$

where $\mathbf{n}_0$ is additive noise and $\boldsymbol{\Upsilon}$ is called the measurement matrix. To obtain accurate estimates of $\boldsymbol{\varphi}$, the measurement matrix $\boldsymbol{\Upsilon}$ should satisfy the restricted isometry property (RIP), i.e. it should be nearly orthonormal when operating on the sparse vector $\boldsymbol{\varphi}$. In following subsections, we derive the observation model, discuss the RIP of the measurement matrix and the accuracy of the reconstructed sparse channel, and evaluate the effect of the system parameters on the performance.

3.1. CS-based channel reconstruction

Following [3], we consider a frame-based UWB MB-OFDM transmission system, in which several known OFDM symbols are placed in a preamble for channel estimation and synchronization, followed by a payload frame containing the OFDM data symbols. During the transmission of the payload frame, the IEEE 802.15.3a standard [11] assumes that the channel remains unchanged.

To estimate the channel impulse response, we consider a preamble, where the frequency domain pilots $\mathbf{X} = [X(0), X(1), \cdots, X(N-1)]$ are selected randomly, and the resulting UWB MB-OFDM symbol is repeated N_p times. We assume a long cyclic prefix is preceding the preamble, to avoid distortion of the preamble due to transition effects at the start of the preamble, and no guard interval is added between the pilot OFDM symbols. Hence, the time-domain samples of each of the N_p pilot OFDM symbols can be written as:

$$x(k+iN) = x(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} e^{j2\pi \frac{kn}{N}} X(n) \tag{9}$$

with $k = 0, \ldots, N-1$ and $i = 0, \ldots, N_p - 1$. After transmitting the preamble over the channel with impulse response $\mathbf{h}$, we obtain the received sequence:

$$y(m) = \sum_{i=0}^{N_p-1} \sum_{k=0}^{N-1} x(k) h(m-k-iN) + w(m) \tag{10}$$

where $w(m)$ is zero-mean additive white Gaussian noise with variance N_0, and $h(l) = 0$ for $l < 0$ or $l \geq L_{taps}$, with L_{taps} the number of taps within the total CIR duration. In this paper, we restrict our observation to the NN_p samples from the pilot preamble to avoid interference with the subsequent data symbols. We assume this preamble is shorter than the total CIR length, i.e. $NN_p < L_{taps}$. However, in order to allow accurate reconstruction of the channel, the preamble must be long enough to capture the contributions of the majority of the dominant components of the channel. As a result, the observation vector (10) can be rewritten as

$$\mathbf{y} = \boldsymbol{\Phi}\mathbf{h} + \mathbf{w}, \tag{11}$$

where $\mathbf{h} = [h(0), h(1), \cdots, h(L_{taps}-1)]^T$, and the elements of the $NN_p \times L_{taps}$ measurement matrix $\boldsymbol{\Phi}$,[1] which are obtained by substituting (9) into (10), are given as

$$\Phi_{m,m'} = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X_{\lfloor * \rfloor \frac{m-m'}{N}}(n) e^{j2\pi \frac{n(m-m')}{N}} \tag{12}$$

with $\lfloor x \rfloor$ the floor of x. To reconstruct the channel, we define a $L_{taps} \times M_{taps}$ dictionary $\boldsymbol{\Psi}$, where $M_{taps} < L_{taps}$ corresponds to an interval containing all dominant channel components. If the dictionary size M_{taps} is too small, not all dominant components will be recovered, but if it is selected too large, noisy samples will affect the ability to properly reconstruct the channel and at the same time the complexity of the algorithm will increase. Hence, the optimal value of M_{taps} must be determined by the receiver. We assume $M_{taps} \geq NN_p$ and $\boldsymbol{\Psi} = [\mathbf{I}\ \mathbf{0}]^T$, where $\mathbf{I}$ is the $M_{taps} \times M_{taps}$ identity matrix. Hence, we restrict our attention to the first M_{taps} taps of the L_{taps} taps of the CIR, which is a reasonable assumption as the tail of the CIR in the considered channel models contain only close-to-zero taps. To reconstruct the channel, we write the CIR $\mathbf{h}$ as a linear combination of $M_{taps} \ll L_{taps}$ basis vectors from the dictionary $\boldsymbol{\Psi}$:

$$\mathbf{h} = \boldsymbol{\Psi}\boldsymbol{\xi} \tag{13}$$

where $\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_{M_{taps}}]^T$ is a $M_{taps} \times 1$ vector. Because the channel is sparse, only K_s of the M_{taps} components of $\boldsymbol{\xi}$ will have

[1] As can be observed in (12), the measurement matrix $\boldsymbol{\Phi}$ is a Hermitian matrix containing the pilots $\mathbf{X}$ and components from the Fourier kernel. Hence, the measurement matrix can in general not be written as a partial Fourier matrix. While it is shown in the literature that, with exponentially high probability, the partial Fourier matrix satisfies the RIP, assuming the number of measurements is nearly linear in the sparsity level, the RIP characteristics of the matrix $\boldsymbol{\Phi}$ will not be straightforward to show, as in general, this proof is a strongly NP-hard problem [23,24]. Although we are not able to prove that the matrix $\boldsymbol{\Phi}$ is RIP, we will demonstrate in the numerical results section that this measurement matrix can be used to accurately estimate the CIR.

non-zero value. Therefore, we can rewrite $\boldsymbol{\xi}$ as

$$\boldsymbol{\xi} = \mathbf{B}\boldsymbol{\theta} \tag{14}$$

where $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_{K_s}]^T$ is the $K_s \times 1$ vector of parameters to be estimated, while the selection matrix $\mathbf{B}$ determines the positions of the K_s non-zero channel taps, i.e.

$$B_{i,j} = \begin{cases} 1 & \text{if } \; i = v(j) \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

where $\mathbf{v} = [v(1), v(2), \cdots, v(K_s)]$ is the $K_s \times 1$ vector with the positions of the K_s dominant taps. Substituting (14) into (13), we obtain:

$$\mathbf{h} = \boldsymbol{\Psi}\mathbf{B}\boldsymbol{\theta} \tag{16}$$

Note that $\boldsymbol{\theta}$ can be estimated using standard channel estimation techniques such as LS or MMSE, however, the main drawback of these approaches is that if the CIR length L_{taps} exceeds the preamble length NN_p, as in the problem at hand, the accuracy of the channel estimation is degraded. To overcome this issue, CS can be used. It is shown in [25,26] that the channel estimation accuracy of CS outperforms that of LS and MMSE methods when $L_{taps} > NN_p$. Moreover, the complexity of the CS method is lower than that of the MMSE approach. Hence, provided that the measurement matrix $\boldsymbol{\Phi}$ is incoherent with the dictionary $\boldsymbol{\Psi}$ (which is the case in the problem at hand), the CIR can be estimated with high reliability through solving the following well-known convex l_1-norm optimization problem:

$$\hat{\boldsymbol{\xi}} = \min \|\boldsymbol{\xi}\|_1 \quad \text{s.t.} \quad \mathbf{y} = \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{\xi}. \tag{17}$$

This convex optimization problem can be solved using linear programming techniques like subspace pursuit (SP), orthogonal match pursuit (OMP) and compressive sampling matching pursuit (CoSaMP). Let us take a closer look at the performance of the CS estimation. Assume the CIR has non-zero components at positions $\mathbf{v}$. The CS algorithm first has to estimate the positions $\hat{\mathbf{v}} = [\hat{v}(1), \cdots, \hat{v}(K_s)]$ of the dominant channel taps, and then it needs to estimate the values $\hat{\mathbf{h}}_{\hat{\mathbf{v}}}$ of the dominating channel taps. To this end, we define, for given M_{taps} and K_s, the matrix $\mathbf{A} = \boldsymbol{\Phi}\boldsymbol{\Psi} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_{M_{taps}}]$, and the submatrix $\mathbf{A}_{\hat{\mathbf{v}}} = [\boldsymbol{a}_{\hat{v}(1)}, \cdots, \boldsymbol{a}_{\hat{v}(K_s)}]$ of the matrix $\mathbf{A}$. The LS estimate of the dominant channel taps at positions $\hat{\mathbf{v}}$ is given by

$$\hat{\mathbf{h}}_{\hat{\mathbf{v}}} = \boldsymbol{\Psi}(\mathbf{A}_{\hat{\mathbf{v}}}^H \mathbf{A}_{\hat{\mathbf{v}}})^{-1} \mathbf{A}_{\hat{\mathbf{v}}}^H \mathbf{y} \tag{18}$$

We assume that, at positions different from $\hat{\mathbf{v}}$, the reconstructed channel taps are set to zero, i.e. $\hat{\mathbf{h}} = [0, \cdots, 0, \hat{h}_{\hat{\mathbf{v}}}(\hat{v}(1)), 0, \cdots, 0, \hat{h}_{\hat{\mathbf{v}}}(\hat{v}(K_s)), 0, \cdots, 0]^T$. Taking this into account, the MSE of $\mathbf{h}$, i.e. $MSE = E[\|(\hat{\mathbf{h}} - \mathbf{h})\|_2^2]$ yields

$$MSE = \mathbf{h}_{\hat{\mathbf{v}}}^H (\mathbf{C}\boldsymbol{\Phi} - \mathbf{I})^H (\mathbf{C}\boldsymbol{\Phi} - \mathbf{I}) \mathbf{h}_{\hat{\mathbf{v}}} + \mathbf{h}_{\mathbf{v}\setminus\hat{\mathbf{v}}}^H \mathbf{h}_{\mathbf{v}\setminus\hat{\mathbf{v}}} + N_0 trace(\mathbf{C}^H\mathbf{C}) \tag{19}$$

where the first and third term originate from the contribution of the reconstructed channel at the positions $\hat{\mathbf{v}}$, and the second contribution from the non-zero channel taps that were not selected in the reconstruction, i.e. at positions $\mathbf{v} \setminus \hat{\mathbf{v}}$. In (19), $\mathbf{C} = \boldsymbol{\Psi}(\mathbf{A}_{\hat{\mathbf{v}}}^H \mathbf{A}_{\hat{\mathbf{v}}})^{-1} \mathbf{A}_{\hat{\mathbf{v}}}^H$ and $\mathbf{h}_{\hat{\mathbf{v}}} = [h(\hat{v}(1)), \cdots, h(\hat{v}(K_s))]^T$. Defining $\mathbf{e} = (\mathbf{C}\boldsymbol{\Phi} - \mathbf{I})\mathbf{h}_{\hat{\mathbf{v}}}$, the first term of (19) can be rewritten as

$$\mathbf{e}^H\mathbf{e} = \mathbf{e}_{\mathbf{v}\cap\hat{\mathbf{v}}}^H \mathbf{e}_{\mathbf{v}\cap\hat{\mathbf{v}}} + \mathbf{e}_{\hat{\mathbf{v}}\setminus\mathbf{v}}^H \mathbf{e}_{\hat{\mathbf{v}}\setminus\mathbf{v}} \tag{20}$$

i.e., the contribution of the positions that are both included in $\mathbf{v}$ and $\hat{\mathbf{v}}$ – or the dominant channel taps that were correctly identified with the CS algorithm – and the contribution from zero channel taps that were incorrectly identified as dominant channel taps ($\hat{\mathbf{v}} \setminus \mathbf{v}$). Substituting (20) in (19), the MSE of $\mathbf{h}$ can be decomposed as

$$MSE = e_h + e_n \tag{21}$$

where the first term $e_h = \mathbf{e}_{\mathbf{v}\cap\hat{\mathbf{v}}}^H \mathbf{e}_{\mathbf{v}\cap\hat{\mathbf{v}}} + \mathbf{h}_{\mathbf{v}\setminus\hat{\mathbf{v}}}^H \mathbf{h}_{\mathbf{v}\setminus\hat{\mathbf{v}}}$ contains a channel-dependent term, originating from the estimation errors in the dominant components due to the CS reconstruction error, and the second term $e_n = \mathbf{e}_{\hat{\mathbf{v}}\setminus\mathbf{v}}^H \mathbf{e}_{\hat{\mathbf{v}}\setminus\mathbf{v}} + N_0 trace(\mathbf{C}^H\mathbf{C})$ contains the contributions from the noise and the estimation errors at time instants that do not contain dominant channel components, but that are captured by the CS algorithm due to noise. Both terms depend on the selection of the parameters M_{taps} and K_s through the matrix $\mathbf{C}$. Preferably, the mismatch related to the compressive sensing must be made as small as possible. To obtain an accurate reconstruction of the channel, we need to select out of the M_{taps} channel components, the K_s taps with the largest energy, i.e. determine the positions of the non-zeros components of $\mathbf{B}$, and estimate the values of $\boldsymbol{\theta}$ for the selected M_{taps} and K_s. The best performance of channel estimation is obtained when M_{taps} and K_s are matched to the effective duration and the dominant taps of the channel.

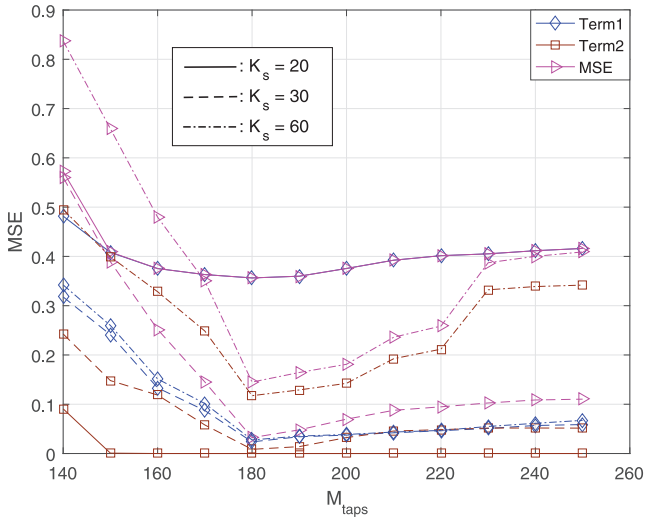
3.2. Effect of the dictionary size and sparsity

To solve the l_1-norm optimization problem discussed in the previous section, we will employ the CoSaMP algorithm [19], as it combines good estimation accuracy with low complexity. However, similarly as for the LS and MMMSE approaches, this algorithm requires the knowledge of the size M_{taps} of the dictionary, which must be matched to the effective duration of the channel, as well as the sparsity K_s, to be able to reconstruct the channel. In this section, we first theoretically analyze the effect of the dictionary size M_{taps} and sparsity K_s, as an improper choice of M_{taps} or K_s can strongly affect the channel estimation performance and thus will influence the bit error rate (BER) performance. Then, some simulations are conducted to verify our analysis.

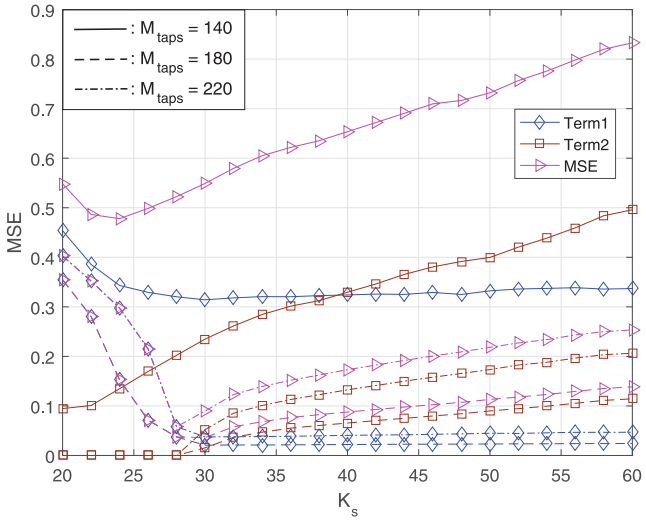
First, we consider the effect of the dictionary size M_{taps} on the performance. On the one hand, if the dictionary size is too small, the search window will not be able to 'catch' all significant channel taps, causing a degradation of the channel estimation performance. On the other hand, if we select the dictionary size too large, not only the complexity of the algorithm will increase, but also the estimation process will include samples having near-zero amplitude, implying noise will start to play a larger role. As a result, increasing the dictionary size will have a detrimental effect on both the performance and the complexity.

Similarly, if the sparsity K_s is too small, not enough significant signal components will be uncovered in the channel reconstruction process. As part of the channel taps with non-negligible energy will not be included in this way, the BER performance will be degraded. However, if the sparsity is too large, the algorithm will start to include signal components with close-to-zero amplitude, indicating that the channel estimation error will increase due to the relatively large effect of the noise on these close-to-zero channel components. Moreover, increasing the sparsity will augment the computational complexity of the algorithm.

The dependency of the MSE on M_{taps} and K_s is illustrated in Fig. 4, in which a pilot preamble with $N = 128$ and $N_p = 1$ is transmitted to reconstruct the CIR for a signal-to-noise ratio (SNR) E_b/N_o of 30dB. The results are averaged over 500 random CM1-based channel realizations with $M_{taps,opt} = 180$ effective taps and $K_{s,opt} = 30$ dominant taps. As explained, the MSE will first reduce when M_{taps} is increased, and slightly increases again when $M_{taps} > M_{taps,opt}$. Decomposing the MSE into the channel-dependent term (first term of (21), Term1 in Fig. 4) and the noise-dependent term (second term of (21), Term 2 in Fig. 4), we observe in Fig. 4(a) that both the channel-dependent term and noise-dependent term grow when M_{taps} reduces, for $M_{taps} < M_{taps,opt}$. When M_{taps} reduces, dominant taps will start to fall outside the interval $[0, M_{taps}]$,



(a)



(b)

Fig. 4. MSE and its constituent terms as function of (a) M_{taps} for different K_s and (b) K_s for different M_{taps}.

implying those taps cannot be reconstructed by the algorithm, so the mismatch between the true and reconstructed channel increases, i.e. the channel-dependent term grows. At the same time, as not enough dominant taps will be present in the interval $[0, M_{taps}]$, the algorithm will start to reconstruct noise-dominated taps, implying the noise-dependent term also increases. This latter effect is larger when K_s is larger, as the algorithm tries to reconstruct more close-to-zero taps. When $M_{taps} > M_{taps,opt}$, the increase of the MSE as function of M_{taps} is mainly due to the increase of the noise-dependent term, as the channel-dependent term is quasi-independent of M_{taps}, because the dominant taps are mainly contained in the first part of the interval, i.e. in $[0, M_{taps,opt}]$.

Further, we stated that when K_s is too small, some dominant channel taps will not be reconstructed, while when K_s is too large, noise will start to play a larger role. This is observed in Fig. 4(b). When $K_s < K_{s,opt}$, the MSE is dominated by the channel-dependent term, while this term becomes independent of K_s when $K_s > K_{s,opt}$, i.e. all dominant taps will be reconstructed. On the other hand, the noise-dependent term grows with K_s, as more noise-dominated taps will be taken into account.

4. Parameter estimation

The dictionary size M_{taps} and the sparsity K_s both affect the precision of channel estimation. Hence, knowledge about the parameters M_{taps} and K_s is indispensable. However, in realistic scenarios, prior information about these two parameters is often not available, so we will estimate these parameters based on the preamble. In this section, we first discuss the simplex method to jointly estimate M_{taps} and K_s. As the complexity of the simplex method is very high, we then propose a simpler, sub-optimal method to estimate M_{taps} and K_s. We show that the performance degradation of our method compared to the simplex method is small. Finally, we compare the complexity of the proposed method with that of the simplex method and the MMSE estimator.

4.1. Joint estimation

To jointly optimize M_{taps} and K_s, we need to solve the following optimization problem:

$$< M_{taps,opt}, \quad K_{s,opt} >= \operatorname{argmin}_{M_{taps},K_s} \|\mathbf{\Phi\Psi}\hat{\boldsymbol{\xi}} - \mathbf{y}\|_2 \quad \text{s.t.} \quad \mathbf{y} = \mathbf{\Phi\Psi}\boldsymbol{\xi}. \tag{22}$$

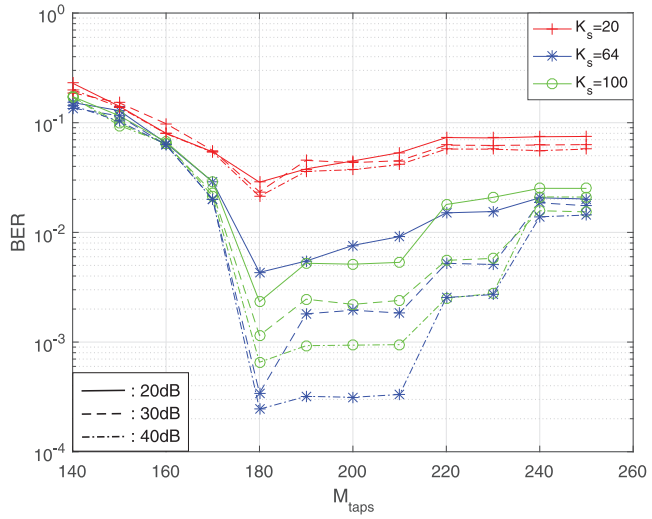
which can be solved with the well-known simplex algorithm. This method can accurately estimate M_{taps} and K_s, as will be illustrated in Section 5. Although it is stated in [27] that the simplex algorithm in general needs polynomial time, in the worst case, the simplex algorithm applied to the problem at hand has complexity $\mathcal{O}\{{M_{taps,opt}}^2 K_{s,opt} \log_2(M_{taps,opt}) \log_2(K_{s,opt})\}$. As this complexity is still very high, the simplex algorithm is unsuitable for practical implementation in UWB MB-OFDM systems.

4.2. Adaptive CE method

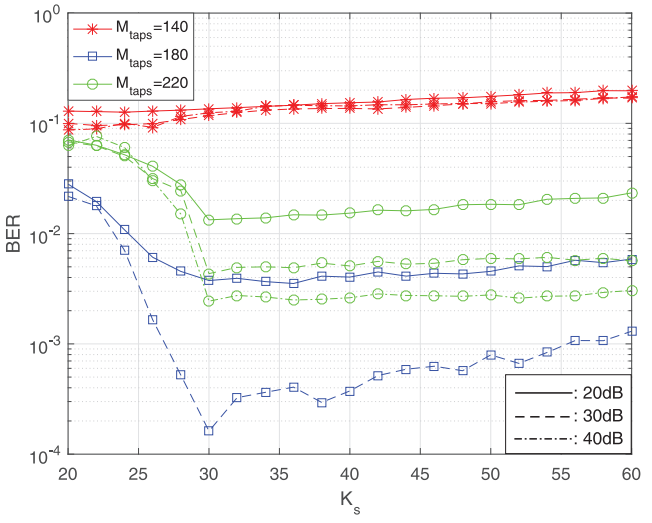
As jointly optimizing K_s and M_{taps} is a challenging task, we first take a look at the effect of both parameters on the BER performance, to be able to reduce the computational burden of the optimization. In Fig. 5, we show the BER of the pilot sequence as function of the dictionary size M_{taps} and sparsity K_s, with $N = 128$ and $N_p = 1$. Comparing Figs. 4(a) and 5(a), we observe that the minimum of the BER coincides with the minimum of the MSE, i.e. at $M_{taps,opt} = 180$, if K_s is sufficiently large. This optimal value of M_{taps} for the BER does neither depend on the sparsity K_s nor on the SNR E_b/N_o. On the other hand, from Fig. 5(b), where the BER is shown as function of the sparsity K_s, we find that when M_{taps} is smaller than the optimum $M_{taps,opt} = 180$ found in Fig. 5(a), i.e. for $M_{taps} = 140$, the BER is largely independent of the value of K_s, while when $M_{taps} \geq M_{taps,opt}$, the BER shows a clear minimum at an intermediate value of K_s, i.e. at $K_{s,opt} = 30$. As the optimal value of M_{taps} is not prior known at the receiver, this implies that we need to first estimate the dictionary size, to avoid that no optimum value for the sparsity can be found. Similar results were obtained for the other channel models. Hence, in the following, we propose an estimation algorithm for M_{taps} and K_s, where in the first phase, the optimal value of M_{taps} is determined, and in the second phase, the optimal value of K_s. In both phases, we employ a dynamic search algorithm.

4.2.1. Algorithm 1: $\mathrm{M_{taps}}$ optimization

First, we will describe the algorithm to estimate M_{taps}. In this algorithm, we need an initial value for K_s. First, we note that, when $K_s > K_{s,opt}$, the optimal value $M_{taps,opt}$ becomes (quasi-) independent of K_s. Further, we notice from Fig. 5 that selecting $K_s > K_{s,opt}$ will have a smaller effect on the BER than selecting $K_s < K_{s,opt}$. Therefore, we select $K_{s,initial} > K_{s,opt}$. As no prior knowledge is available, we set $K_{s,initial}$ to the maximum possible value that can be estimated with the preamble, i.e. $K_{s,initial} = NN_p$. To find $M_{taps,opt}$, we



(a)



(b)

Fig. 5. BER performance as function of (a) M_{taps} for different K_s and (b) K_s for different M_{taps}.

define an initial search interval $[M_{\min},\ M_{\max}]$. For the lower limit, we set $M_{\min} = NN_p$. To motivate this lower limit, we note that in practice, if the length of the pilot preamble is shorter than the effective CIR length, the channel can not be estimated accurately. Hence, we assume that in the design phase, the length of the pilot preamble is selected in a proper way, i.e. that it is able to catch all dominating channel components. Therefore, it is clear that the length NN_p of the pilot preamble is an appropriate lower limit for M_{taps}. For the upper limit, we select $M_{\max} = 5NN_p$, as it is shown in [28] that if the length NN_p of the pilot sequence is smaller than 20% of the CIR length, the CIR cannot be reconstructed accurately. Note that if the optimal value $M_{taps,opt}$ falls outside this initial search interval, the algorithm can adapt the search range automatically (see lines 10–13, Algorithm 1) . However, for both simulated channels and measured channels, we did not encounter any case where the optimal $M_{taps,opt}$ was located outside this initial search interval, meaning the search interval was selected properly. Within the initial search interval, we select N_b equidistant values for M_{taps}, i.e. with step size $M_{step} = \lfloor * \rfloor (M_{\max} - M_{\min})/N_b$. For

Algorithm 1 M_{taps} estimation algorithm.

1: Initialization: Set $K_{s,initial} = NN_p$, searching block $[M_{\min},\ M_{\max}]$, $M_{\min} = NN_p$, $M_{\max} = 5NN_p$, pilot sequence $\mathbf{X}$, received pilot sequence $\mathbf{y}$, the number N_b of test values, $M_{step} = +\infty$ and $M_{step,min} = 1$.
2: **while** $M_{step} > M_{step,min}$ **do** % *outer loop*
3: $M_{step} \Leftarrow \lfloor (M_{\max} - M_{\min})/N_b \rfloor$;
4: **for** $m = 0 : N_b$ **do** % *inner loop*
5: $M_{taps} \Leftarrow M_{\min} + m * M_{step}$;
6: Obtain $\hat{\mathbf{h}}$ with CoSaMP using M_{taps}, $K_{s,initial}$, $\mathbf{X}$ and $\mathbf{y}$;
7: Calculate BER_m with $\hat{\mathbf{h}}$;
8: **end for**
9: $m_{opt} \Leftarrow arg\min_m BER_m$
10: **if** $m_{opt} = 0$ or $m_{opt} = N_b$ **then**
11: $M_{taps,opt} \Leftarrow M_{\min} + m_{opt} * M_{step}$;
12: $M_{\max} \Leftarrow M_{taps,opt} + NN_p$;
13: $M_{\min} \Leftarrow M_{taps,opt} - NN_p$;
14: **else**
15: $M_{taps,opt} \Leftarrow M_{\min} + m_{opt} * M_{step}$;
16: $M_{\max} \Leftarrow M_{taps,opt} + M_{step}$;
17: $M_{\min} \Leftarrow M_{taps,opt} - M_{step}$;
18: **end if**
19: **end while**
20: Output $M_{taps,opt}$.

each of the N_b selected values M_{taps}, we reconstruct the CIR with the CoSaMP algorithm, and use the reconstructed channel to recover the data of the pilot sequence and to compare the resulting bits with the known pilot symbols to obtain the BER, which serves as the optimization criterion in our algorithm. From these N_b test values, we select the value $M_{taps,opt}$ that minimizes the BER. We tighten the search interval around the found value of $M_{taps,opt}$, i.e. $[M_{taps,opt} - M_{step},\ M_{taps,opt} + M_{step}]$, reduce the step size with a factor $N_b/2$ (see line 3 of Algorithm 1), and continue the search procedure until the step size becomes smaller than or equal to the threshold $M_{step,min}$, which we chose $M_{step,min} = 1$ in our simulations.

4.2.2. *Algorithm 2:* K_s *estimation*

The dynamic window search algorithm to find $K_{s,opt}$, which is shown in Algorithm 2, is similar to the algorithm to find $M_{taps,opt}$. The initial search interval for K_s is set to $[K_{\min},\ K_{\max}] = [1,\ NN_p]$, i.e. the minimum and maximum K_s that can be esti-

Algorithm 2 K_s estimation algorithm.

1: Initialization: Set $M_{taps,opt}$, searching block $[K_{\min}\ K_{\max}]$, $K_{\min} = 1$, $K_{\max} = NN_p$, pilot sequence $\mathbf{X}$, received pilot sequence $\mathbf{y}$, the number N_b of test values, $K_{step} = +\infty$ and $K_{step,min} = 1$.
2: **while** $K_{step} > K_{step,min}$ **do** % *outer loop*
3: $K_{step} \Leftarrow \lfloor (K_{\max} - K_{\min})/N_b \rfloor$;
4: **for** $k = 0 : N_b$ **do** % *inner loop*
5: $K_s \Leftarrow K_{\min} + k * K_{step}$
6: Obtain $\hat{\mathbf{h}}$ with CoSaMP using $M_{taps,opt}$, K_s, $\mathbf{X}$ and $\mathbf{y}$;
7: Calculate BER_k with $\hat{\mathbf{h}}$;
8: **end for**
9: $k_{opt} \Leftarrow arg\min_k BER_k$
10: $K_{s,opt} \Leftarrow K_{\min} + k_{opt} * K_{step}$;
11: $K_{\max} \Leftarrow K_{s,opt} + K_{step}$;
12: $K_{\min} \Leftarrow K_{s,opt} - K_{step}$;
13: **end while**
14: Output $K_{s,opt}$, CIR $\hat{\mathbf{h}}$ estimated with $M_{taps,opt}$ and $K_{s,opt}$.

mated with a preamble length NN_p. Similarly as in Algorithm 1, we select N_b equidistant test values for K_s, with step size $K_{step} = \lfloor * \rfloor (K_{\max} - K_{\min})/N_b$, and reconstruct the CIR with the selected value of K_s and $M_{taps,opt}$ from Algorithm 1, and compute the resulting BER of the pilot sequence. We gradually refine our search interval until the step size K_{step} is smaller than or equal to the threshold $K_{step,min} = 1$. The outputs of this algorithm are $K_{s,opt}$ and the reconstructed CIR.

4.3. Complexity analysis

To show that the proposed algorithm is suitable for practical implementation, we evaluate the computational complexity of the algorithm. This computational complexity, which is expressed in terms of the number of complex multiplications (NCM), is compared to the complexity of the MMSE estimator. The main share of the computations in the proposed algorithm stems from the CoSaMP algorithm that is used to reconstruct the CIR for each combination of M_{taps} and K_s. From [19], the number of complex multiplications required in the CoSaMP algorithm equals $\mathcal{O}\{M_{taps}\log_2(M_{taps})\log_2(K_s)\}$. Hence, this complexity depends on the considered M_{taps} and K_s. In the M_{taps} optimization algorithm, given in Algorithm 1, K_s is fixed to $K_{s,initial}$, but M_{taps} changes during the course of the optimization. As in this optimization process, the dictionary size M_{taps} will converge to $M_{taps,opt}$, we approximate the complexity of the CoSaMP algorithm by $\mathcal{O}\{M_{taps,opt}\log_2(M_{taps,opt})\log_2(K_{s,initial})\}$. Within each inner loop, N_b+1 values of M_{taps} are tested, and the inner loop is executed $\mathcal{O}\{\log_{N_b}(4NN_p)\}$ times, leading to the complexity $\mathcal{O}\{(N_b+1)M_{taps,opt}\log_2(M_{taps,opt})\log_2(K_{s,initial})\log_{N_b}(4NN_p)\}$. Similarly, in the K_s-optimization algorithm, where M_{taps} is fixed to $M_{taps,opt}$, the complexity is approximated by $\mathcal{O}\{M_{taps,opt}\log_2(M_{taps,opt})\log_2(K_{s,opt})\}$, as during the optimization process, K_s grows closer to $K_{s,opt}$. Further, per inner loop, N_b+1 values of K_s are tested and the inner loop is executed $\mathcal{O}\{\log_{N_b}(NN_p)\}$ times, resulting in a complexity $\mathcal{O}\{(N_b+1)M_{taps,opt}\log_2(M_{taps,opt})\log_2(K_{s,opt})\log_{N_b}(NN_p)\}$. The total complexity of the algorithm therefore equals $\mathcal{O}\{(N_b+1)M_{taps,opt}\log_2(M_{taps,opt})\cdot[\log_2(K_{s,initial})\log_{N_b}(4NN_p)+\log_2(K_{s,opt})\log_{N_b}(NN_p)]\}$.

To compare the complexity of the proposed algorithm with the complexity of the MMSE estimator and the simplex algorithm, which are $\mathcal{O}\{(NN_p)^3\}$ and $\mathcal{O}\{{M_{taps,opt}}^2 K_{s,opt}\log_2(M_{taps,opt})\log_2(K_{s,opt})\}$ respectively, we consider the case where $M_{taps,opt} = 5NN_p$, $K_{s,opt} = NN_p$ and $N_b = 5$. As in most situations, $M_{taps,opt}$ and $K_{s,opt}$ will be smaller than these values, the true complexity will be smaller than the complexity shown in Fig. 6. As can be observed in the figure, compared with MMSE estimator and the proposed method, the high complexity of simplex method makes it impracticable in channel estimation. Comparing the MMSE estimator and our method, we find that when $NN_p > 112$, the proposed method has lower complexity than the MMSE method. Moreover, taking into account that the worst case values of M_{taps} and K_s were taken to compute the computational complexity of our method, in practical situations, the complexity reduction compared to the MMSE approach will be much larger than shown in the figure.

5. Numerical results

In this section, we verify the performance of the proposed adaptive CE method. The channel models (CMs) used in our simulations are based on the UWB communication environments and propagation scenarios considered in the IEEE 803.15.3a standard [11] (see Table 1 for the parameters of these CMs). For each of the four considered CMs, 500 random channel realizations are generated, each having the same channel length L_{taps}, with $L_{taps} =$ 600, 730, 1300, 2150 for CM 1–4, corresponding to $K_s = 30,\ 45,\ 92$ and 163 dominant non-zero taps, that are distributed over the first $M_{taps} = 180,\ 202,\ 341$ and 698 taps of the channel, respectively. In the simulations, we set the bandwidth to 528 MHz, in the frequency band 5280–5808 MHz, which corresponds to band #5 in [29]. Further, we take the number of subcarriers equal to 128, 256, 512 and 1024, which corresponds to a carrier spacing of 4.1251 MHz, 2.0625 MHz, 1.0313 MHz and 0.5157 MHz, respectively. The sampling rate used in our simulations equals the bandwidth. An overview of the parameters is given in Table 2.

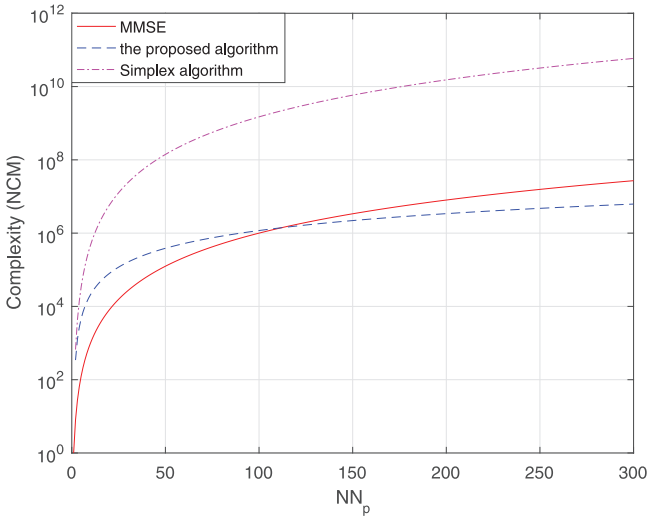


Fig. 6. Complexity comparisons of MMSE, the simplex algorithm and the proposed algorithm.

Table 2
Parameters of the MB-OFDM system.

Bandwidth (MHz)	528			
RF carrier frequency (MHz)	5544			
Frequency bandwidth (MHz)	5280–5808			
Number of subcarriers N	128	256	512	1024
Subcarrier spacing Δf (MHz)	4.1251	2.0625	1.0313	0.5157
Sampling rate (MHz)	528			

We first assess the probability of miss detection of M_{taps} and K_s, i.e. $P_{miss,\ M_{taps}} = P(M_{taps,opt} \neq M_{taps,real})$ and $P_{miss,K_s} = P(K_{s,opt} \neq K_{s,real})$. To this end, we consider the case of CM1, and transmit a pilot preamble of length $NN_p = 128$, in which $N = 128$ and $N_p = 1$. In Fig. 7, the probability of miss detection of M_{taps} and K_s is shown for the proposed algorithm, and compared with those for the simplex algorithm which is implemented using the function of `fminsearch` in Matlab. The figure demonstrates that when E_b/N_o increases, the probability of miss detection of both M_{taps} and K_s reduces, and drops below 5% when E_b/N_o is sufficiently large. Hence, both algorithms can determine the parameters K_s and M_{taps} with high accuracy. Although the proposed algorithm has a slightly higher probability of miss detection, the much lower complexity of the proposed algorithm makers it more suitable for practical use. We also evaluated the probability of miss detection for the other channel models CM 2–4, and the results are similar to that of CM1.

Next, we evaluate the performance of the proposed channel estimator and compare the results with the performance of OMP, SP and CoSaMP. For each of the considered algorithms, we estimate the CIR for 500 channel realizations of CM1, and compute the normalized mean-squared error (NMSE) $E[\|(\hat{\mathbf{h}}-\mathbf{h})\|_2^2\ /\|(\mathbf{h})\|_2^2]$. For the OMP, SP and CoSaMP algorithms, we assume no prior knowledge of M_{taps} and K_s is available, and we set $M_{taps} = 256$ and $K_s =$ 60 for these algorithms. As can be observed in Fig. 8, the proposed

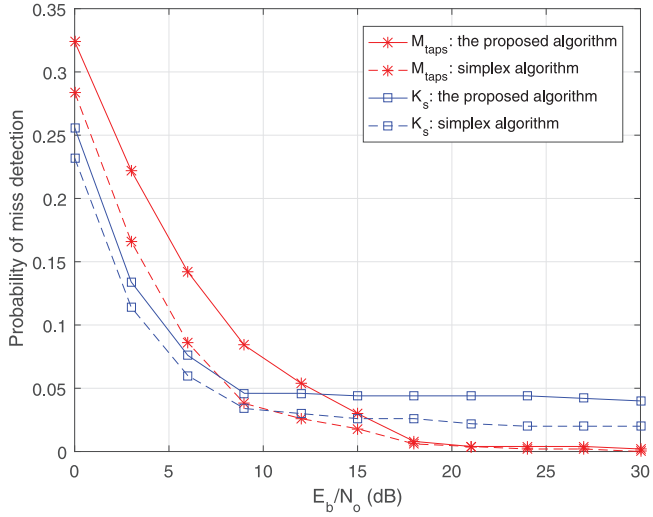


Fig. 7. Probabilities of miss detection for M_{taps} and K_s.

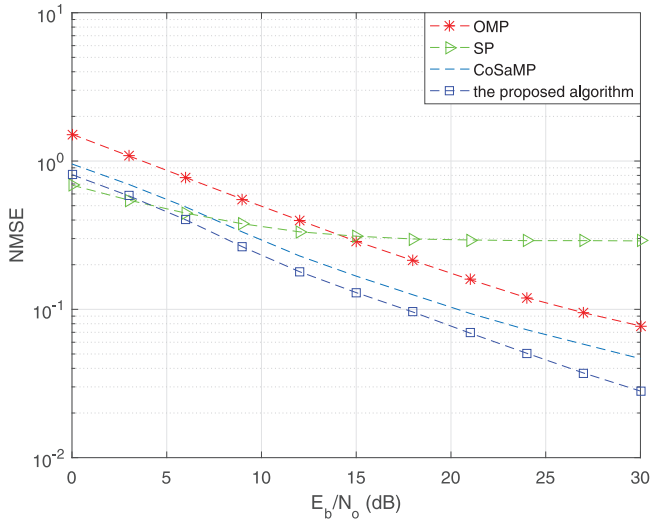


Fig. 8. NMSE of the proposed method and other CS reconstruction methods.

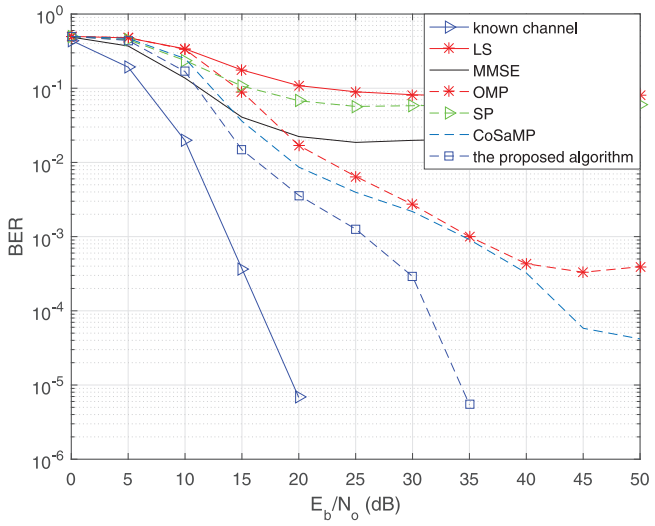
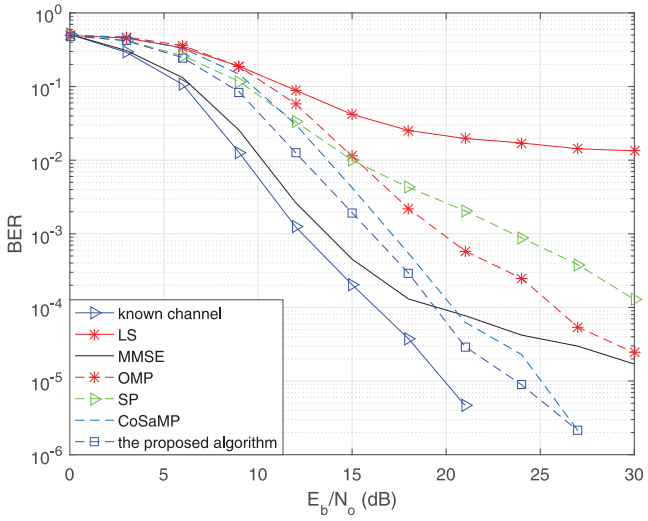


Fig. 9. BER performance of the proposed method compared to other selected methods for CM1.

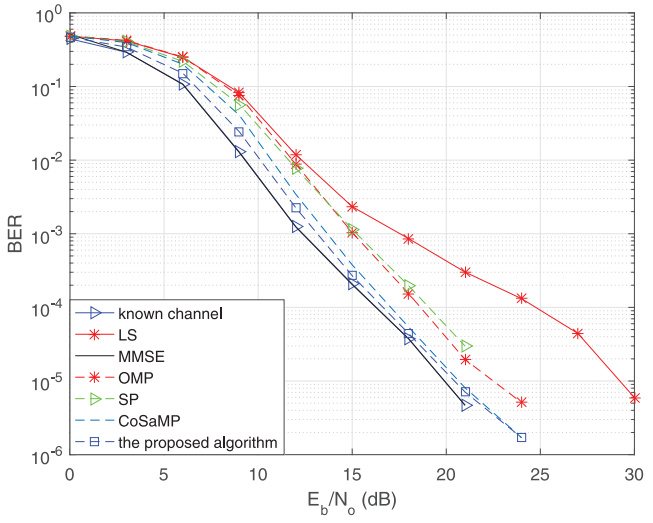
algorithm outperforms the other algorithms. This can be attributed to the optimized dictionary size and sparsity in our method, resulting in a more accurate reconstruction of the CIR, especially at higher E_b/N_o.

To evaluate the BER performance of the proposed estimator, we transmit for each channel realization a pilot preamble of NN_p pilot symbols, followed by 5000 OFDM symbols, each containing 1024 data symbols, where the data is modulated using quadrature phase-shift keying (QPSK). The BER of CM i is obtained by averaging the BER of the transmitted data over 500 channel realizations of CM i. The BER performance of the proposed algorithm is compared with the performance of other CS methods, i.e. SP, OMP and CoSaMP, traditional CE methods, i.e. LS and MMSE, and the case where the channel is known. For the SP, OMP and CoSaMP methods, the same values for M_{taps} and K_s are used as mentioned for Fig. 8. For CM1, the results are shown in Fig. 9, assuming $N = 128$ and $N_p = 1$. Taking into account that the length of this pilot preamble is shorter than the effective CIR length, which was 180 in our simulation, the channel estimators are not able to extract all dominant CIR components. This results in a degradation of the BER compared to the case where the channel is perfectly known at the receiver. Comparing the BER results, we observe that the proposed method outperforms all other CEs, although the gap with the BER of the known channel is still relatively large. By increasing the pilot preamble length, by e.g. selecting $N_p = 4$ and $N_p = 8$, this gap will become smaller (see Fig. 10), as more CIR components will be found. For $N_p = 8$, several of the estimators, including the proposed one, exhibit a performance close to the ideal case with known channel. However, the required pilot preamble is quite long, which will limit the data throughput and thus its practical applicability. Therefore, this pilot preamble will be shortened, implying the proposed method will be an excellent solution to estimate the channel in a practical implementation. Furthermore, we show the BER performance for the other channel models. In Fig. 11, the BER performance of the proposed method and other selected methods is shown, assuming $N_p = 1,\ 2$ and 4 pilot symbols of length $N = 128$ are used for CM2, CM3 and CM4, respectively. Our proposed method found for the considered cases that the optimal dictionary size was $M_{taps,opt} = 202,\ 341$ and 698, and the optimal sparsity equalled $K_{s,opt} = 45,\ 92$ and 163, for CM2, CM3 and CM4, respectively. We observed in our simulations that these numbers for $M_{taps,opt}$ and $K_{s,opt}$ are in accordance with the average effective duration and the number of dominant taps of the CIRs. With these parameter settings, we are able to recover about 85% of the power contained in the CIR. Fig. 11 confirms the results from Fig. 9, i.e. the proposed estimator outperforms the other CE methods when the length of the pilot preamble is shortened for practical reasons, so that not all dominant channel components can be retrieved. Similarly as for CM1, when the length of the pilot preamble would be increased, the difference between the performance of the considered CEs and that of the case with known channel will become smaller.

Finally, we applied the proposed method for the channels, measured with the setup from Section 2.2. In this paper, we just show the results for channels between transmitter and receiver positions R_6, R_{12} and R_{14}, corresponding to a LoS, OLoS and NLoS scenario, and denoted by Channel 6, Channel 12 and Channel 14, respectively. The results for the other channels are similar to the results of these three channels. From the measured data, we determined the effective durations of Channel 6, 12 and 14, i.e. they are 168, 241 and 369 respectively, and the values of sparsity are 37, 61 and 91. We select $N_p =$ 1,1 and 2 pilot symbols of length $N =$ 128 to reconstruct Channel 6, 12 and 14, respectively. Further, to



(a)



(b)

Fig. 10. BER performance of the proposed method compared to other selected methods for different lengths of the pilot preamble: (a) $NN_p = 512$ (b) $NN_p = 1024$.

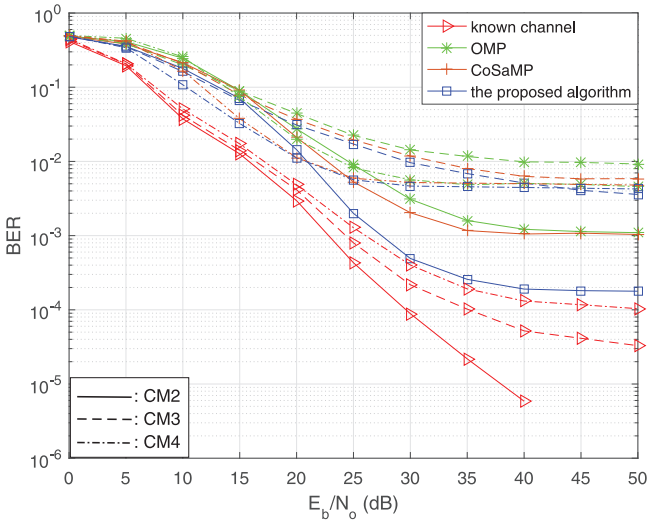


Fig. 11. BER performance of the proposed method compared to other selected methods for CM2, CM3 and CM4.

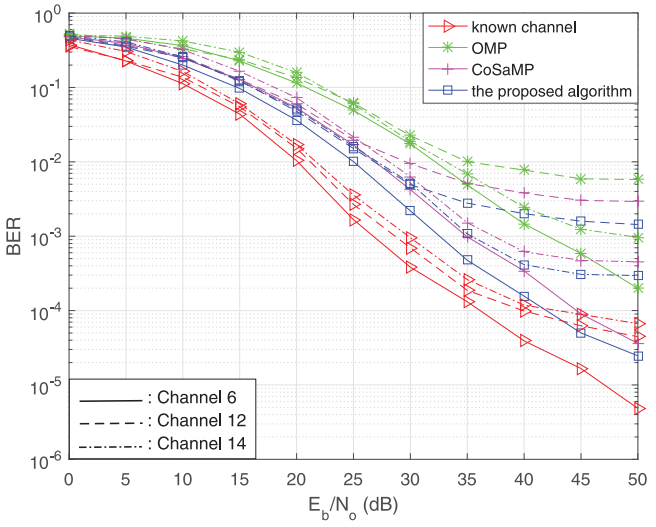


Fig. 12. BER performance for real, measured channels of channel 6, channel 12 and channel 14.

evaluate the BER performance of our method, we transmitted 5000 OFDM symbols modulated by QPSK. The average BERs of the three channels are shown in Fig. 12. Compared with OMP and CoSaMP, the proposed method achieves a lower BER. Although the pilot sequence has the same length for Channel 6 and 12, the effective duration and sparsity of Channel 6 are smaller than those of Channel 12, so the BER performance of Channel 6 is much better. Moreover, thanks to the longer pilot sequence, the average BER of Channel 14 is lower than that of Channel 12, even though Channel 14 has a longer effective duration and more dominant taps. Note that we can improve the BER for Channel 12 by using the same pilot sequence length as for Channel 14. In that case, the BER of Channel 12 turned out to be better than for Channel 14 (results not shown in the figure). Similar to the results of the simulated channels, the performance difference between the CS-based CE methods and the known channel will be reduced when the length of pilot preamble increases.

6. Conclusion

In this paper, we propose an adaptive CS-based parameter estimation algorithm for UWB MB-OFDM systems. Our method extends the CoSaMP method by also estimating the dictionary size M_{taps} and sparsity K_s, needed to accurately estimate the channel. Although state-of-the-art estimators are able to achieve close-to-optimal performance when the pilot preamble length is sufficiently long to extract all CIR components, such pilot preamble lengths are often not suitable for practical implementation as they limit the data throughput. Therefore, in practice the pilot preamble is often shortened. However, this results in a degradation of the system performance. We show in this paper that in such a situation, our method outperforms state-of-the-art channel estimators considerably, as the estimated $M_{taps,opt}$ and $K_{s,opt}$ better match the effective duration and the sparsity of the CIR. Moreover, as the proposed method has low complexity, it is suitable for practical implementation. In this paper, we not only restricted our attention to theoretical channel models, we also evaluated our algorithm for measured channels obtained with the measurement setup described in this paper. The results for these measured channels are

similar to the results of the theoretical channel models, meaning that our method can be employed in practical scenarios.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by a Belgian EOS grant with project grant number EOS30452698, the Flemish fund for Scientific Research (FWO) and the National Natural Science Foundation of China with grant number 61701531.

References

[1] P. Cheng, Z. Chen, F. de Hoog, C.K. Sung, Sparse blind carrier-frequency offset estimation for OFDMA uplink, IEEE Trans. Commun. 64 (12) (2016) 5254–5265.
[2] Z. Wang, Y. Xin, G. Mathew, X. Wang, A low-complexity and efficient channel estimator for multiband OFDM-UWB systems, IEEE Trans. Veh. Technol. 59 (3) (2010) 1355–1366.
[3] A. Batra, Multi-band OFDM physical layer proposal for IEEE 802.15 task group 3a, IEEE P802. 15-03/268r3 (2004).
[4] Y.G. Li, A.F. Molisch, J. Zhang, Practical approaches to channel estimation and interference suppression for OFDM-based UWB communications, IEEE Trans. Wirel. Commun. 5 (9) (2006) 2317–2320.
[5] J.-C. Lin, Least-squares channel estimation for mobile OFDM communication on time-varying frequency-selective fading channels, IEEE Trans. Veh. Technol. 57 (6) (2008) 3538–3550.
[6] L. Deneire, P. Vandenameele, L. Van Der Perre, B. Gyselinckx, M. Engels, A low-complexity ML channel estimator for OFDM, IEEE Trans. Commun. 51 (2) (2003) 135–140.
[7] M. Morelli, U. Mengali, A comparison of pilot-aided channel estimation methods for OFDM systems, IEEE Trans. Signal Process. 49 (12) (2001) 3065–3073.
[8] J.L. Paredes, G.R. Arce, Z. Wang, Ultra-wideband compressed sensing: channel estimation, IEEE J. Sel. Top. Signal Process. 1 (3) (2007) 383–395.
[9] M. Başaran, S. Erküçük, H.A. Çırpan, Compressive sensing for ultra-wideband channel estimation: on the sparsity assumption of ultra-wideband channels, Int. J. Commun. Syst. 27 (11) (2014) 3383–3398.
[10] A.F. Molisch, K. Balakrishnan, C.-C. Chong, S. Emami, A. Fort, J. Karedal, J. Kunisch, H. Schantz, U. Schuster, K. Siwiak, Ieee 802.15. 4a channel model-final report, IEEE P802 15 (04) (2004) 0662.
[11] J. Foerster, Channel modeling sub-committee report final, IEEE P802. 15 Working Group for Wireless Personal Area Networks (WPANs), IEEE P802. 15-02/490r1-SG3a, 2003.
[12] E.J. Candès, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inf. Theory 52 (2) (2006) 489–509.
[13] D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (4) (2006) 1289–1306.
[14] A.H. Muqaibel, M.T. Alkhodary, Practical application of compressive sensing to ultra-wideband channels, IET Commun. 6 (16) (2012) 2534–2542.
[15] M. Özgör, S. Erküçük, H.A. Çirpan, Bayesian compressive sensing for ultra-wideband channel models, in: IEEE 35th International Conference on Telecommunications and Signal Processing (TSP), 2012, pp. 320–324.
[16] X. Cheng, M. Wang, Y.L. Guan, Ultrawideband channel estimation: a Bayesian compressive sensing strategy based on statistical sparsity, IEEE Trans. Veh. Technol. 64 (5) (2015) 1819–1832.
[17] W. Dai, O. Milenkovic, Subspace pursuit for compressive sensing signal reconstruction, IEEE Trans. Inf. Theory 55 (5) (2009) 2230–2249.
[18] J.A. Tropp, A.C. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, IEEE Trans. Inf. Theory 53 (12) (2007) 4655–4666.
[19] D. Needell, J.A. Tropp, Cosamp: iterative signal recovery from incomplete and inaccurate samples, Appl. Comput. Harmon. Anal. 26 (3) (2009) 301–321.
[20] S. Ji, Y. Xue, L. Carin, Bayesian compressive sensing, IEEE Trans. Signal Process. 56 (6) (2008) 2346–2356.
[21] J.C. Nash, The (Dantzig) simplex method for linear programming, Comput. Sci. Eng. 2 (1) (2000) 29–31.
[22] http://electro-metrics.com/wp-content/uploads/2016/12/6865_New_Version%-PAS.pdf.
[23] A.M. Tillmann, M.E. Pfetsch, The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing, IEEE Trans. Inf. Theory 60 (2) (2013) 1248–1259.
[24] A. Natarajan, Y. Wu, Computational complexity of certifying restricted isometry property, Approximation, Randomization, and Combinatorial Optimization: Algorithms and Techniques, 2014.
[25] P. Cheng, Z. Chen, Y. Rui, Y.J. Guo, L. Gui, M. Tao, Q. Zhang, Channel estimation for OFDM systems over doubly selective channels: a distributed compressive sensing based approach, IEEE Trans. Commun. 61 (10) (2013) 4173–4185.
[26] C. Qi, G. Yue, L. Wu, Y. Huang, A. Nallanathan, Pilot design schemes for sparse channel estimation in OFDM systems, IEEE Trans. Veh. Technol. 64 (4) (2015) 1493–1505.
[27] D.A. Spielman, S.-H. Teng, Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time, J. ACM (JACM) 51 (3) (2004) 385–463.
[28] N. Vaswani, W. Lu, Modified-CS: modifying compressive sensing for problems with partially known support, IEEE Trans. Signal Process. 58 (9) (2010) 4595–4607.
[29] S.A. Ghorashi, B. Allen, M. Ghavami, A.H. Aghvami, An overview of MB-UWB OFDM, in: IET Seminar on Ultra Wideband Communications Technologies and System Design, 2004, pp. 107–110.