Comparison of Basis Pursuit Algorithms for Sparse Channel Estimation in Underwater Acoustic OFDM

Jianzhong Huang*, Christian R. Berger†, Shengli Zhou* and Jie Huang*
*Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269, USA
†Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA

***Abstract*—Recently it has been shown that sparse channel estimation, implemented with orthogonal matching pursuit (OMP) and basis pursuit (BP) algorithms, has impressive performance gains over alternatives that do not take advantage of the channel sparsity, for underwater acoustic (UWA) communications. We in this paper compare the performance and complexity of three popular BP algorithms, namely l1_ls, SpaRSA, and YALL1, using both simulation and experimental data for underwater orthogonal frequency division multiplexing (OFDM) systems with both single and multiple transmitters. We find that all BP solvers achieve similar block-error-rate performance, considerably outperforming OMP. In terms of complexity, both SpaRSA and YALL1 reduce the runtime by about one order of magnitude relative to l1_ls, catching up with OMP. The efficient BP solvers such as SpaRSA and YALL1 are thus appealing to be implemented in real-time underwater OFDM modems.**

I. Introduction

Underwater acoustic (UWA) channels are doubly selective channels, having large delay spread and significant Doppler effects. Channel estimation, which is one critical task in the receiver design, is extremely challenging in the UWA communication context due to the large number of unknowns as a result of large delay and Doppler spread. Exploiting the sparse nature of UWA channels has been recently shown to be highly effective, based on *experimental* results for both single-carrier [1] and multi-carrier [2], [3] underwater transmissions.

Orthogonal matching pursuit (OMP) and other matching pursuit (MP) variants have been tested in [1]–[3], while a basis pursuit (BP) algorithm is also tested in [3]. When referring to BP, we mean the solution to the following convex optimization problem

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{z}\|^2 + \zeta \|\mathbf{x}\|_1 , \quad (1)$$

where $\mathbf{z}$ is the observation vector, $\mathbf{A}$ is the dictionary matrix, and ζ is the regularization parameter. In the literature this formulation is also sometimes referred to as $l_1 - l_2$ or Basis Pursuit Denoising (BPDN). MP and BP algorithms have received extensive attention recently under the paradigm of compressed sensing (CS) (see e.g. [4]–[6]). The comparisons in [3], based on simulation and experimental studies, show that BP has better performance than OMP in the UWA context. However, algorithm complexity was not investigated in [3].

The objective of this paper is to study the performance and complexity issues of different BP solvers in the context of UWA sparse channel estimation. In particular, we focus on three different BP implementations, namely l1_ls [7], which had been used in [3], and two recently published algorithms: SpaRSA [8] and YALL1 [9], that have gained much attention in the CS community. All these algorithms can solve the same problem in Eq. (1), but at different complexities. The algorithms were selected based on two criteria: i) the formulation has to deal with complex values, and ii) the matrix $\mathbf{A}$ does not need to be created explicitly, but $\mathbf{A}\mathbf{b}$ or $\mathbf{A}^T\mathbf{c}$ can be evaluated efficiently.

We compare the BP implementations for two different receivers: i) a receiver for a single-transmitter orthogonal-frequency-division-multiplexing (OFDM) system that explicitly suppresses the intercarrier interference (ICI), developed in [3]; and ii) a receiver for a multi-input multi-output (MIMO) OFDM system that treats the ICI as additive noise, developed in [10]. The ICI-aware receiver [3] can handle channels with large Doppler spread, while the ICI-ignorant receiver [10] works well for MIMO channels with small Doppler spread. Both these receivers are confronted with a large number of unknowns in channel estimation, where for the former the number of unknowns is roughly proportional to the product of channel delay spread and Doppler spread, while in the latter it is proportional to the product of delay spread and the number of transmitters. We use both simulation and real data recorded from the SPACE08 experiment, conducted off the coast of Martha's Vineyard, MA, from Oct. 14 to Nov. 1, 2008.

We find that the performance of all BP solvers are similar, but that there are significant differences in complexity. Specifically, SpaRSA and YALL1 outperform l1_ls by about one order of magnitude in run-time. Furthermore, OMP cannot match the BP performance in terms of the block-error-rate (BLER), and has similar complexity with SpaRSA and YALL1 in the ICI-ignorant receiver, even higher complexity in the ICI-aware receiver. Hence, SpaRSA and YALL1 are appealing candidates for sparse channel estimation in underwater OFDM systems. In the future, we plan to pursue real time DSP-based implementations of SpaRSA and YALL1 along with the modem prototype development as described in [11].

The rest of this paper is organized as follows. Section II presents the system model, Section III specifies the dictionary construction, and Section IV describes the BP algorithms. Section V contains numerical results, and we conclude in Section VI.

This work was supported by the ONR grants N00014-07-1-0805 (YIP), N00014-09-1-0704 (PECASE) and the NSF grant CNS-0721834.

II. System Model

We consider zero-padded (ZP) OFDM as in [3], [10], [12]. Let T denote the OFDM symbol duration and T_g the guard interval for the ZP. The total OFDM block duration is $T' = T + T_g$ and the subcarrier spacing is $1/T$. The kth subcarrier is at frequency

$$f_k = f_c + k/T, \quad k = -K/2, \ldots, K/2-1, \tag{2}$$

where f_c is the carrier frequency and K subcarriers are used so that the bandwidth is $B = K/T$. Let $s[k]$ denote the information symbol to be transmitted on the kth subcarrier. The non-overlapping sets of active subcarriers $\mathcal{S}_A$, and null subcarriers $\mathcal{S}_N$ satisfy $\mathcal{S}_A \cup \mathcal{S}_N = \{-K/2, \cdots, K/2-1\}$; The transmitted signal in passband is given by

$$\tilde{x}(t) = \text{Re}\left\{ \left[\sum_{k\in\mathcal{S}_A} s[k] e^{j2\pi \frac{k}{T} t} q(t) \right] e^{j2\pi f_c t} \right\}, \tag{3}$$

for $t \in [0, T+T_g]$ and $q(t)$ describes the zero-padding operation, i.e., $q(t) = 1$ for $t \in [0, T]$.

The same receiver preprocessing as in [3], [10], [12] can be applied, but we neglect resampling for a more compact notation. Let $z(t)$ be the received baseband waveform and ϵ an estimated residual mean Doppler shift, the output signal on the mth subcarrier is

$$z_m = \frac{1}{T}\int_0^{T+T_g} z(t) e^{-j2\pi\epsilon t} e^{-j2\pi \frac{m}{T} t}\, dt. \tag{4}$$

Due to the Doppler effects, there will generally be ICI as,

$$z_m = \sum_{k\in\mathcal{S}_A} H_{m,k} s[k] + v_m, \tag{5}$$

where v_m is the additive noise and $H_{m,k}$'s specify the ICI pattern. We assume a path-based UWA channel model, as in [3]

$$c(\tau, t) = \sum_{p=1}^{N_p} A_p \delta\left(\tau - \tau_p(t)\right) = \sum_{p=1}^{N_p} A_p \delta\left(\tau - (\tau_p - a_p t)\right), \tag{6}$$

where A_p, τ_p, and a_p are the amplitude, delay, and Doppler scale of the pth path.

A. Single transmitter systems

We first consider the single-transmitter channel estimation. For each receiver (for a more compact notation, we ignore the receiver index), with the path-based UWA channel model introduced above, it follows (see [3]),

$$H_{m,k} = \sum_{p=1}^{N_p} A'_p e^{-j2\pi(f_m+\epsilon)\tau'_p} \varrho_{m,k}(a_p), \tag{7}$$

$$\varrho_{m,k}(a_p) = \frac{\sin\left[\pi\beta_{m,k}(a_p)\right]}{\pi\beta_{m,k}(a_p)} e^{j\pi\beta_{m,k}(a_p)}, \tag{8}$$

$$\beta_{m,k}(a_p) = (k-m) + \frac{a_p f_m - \epsilon}{1+a_p} T. \tag{9}$$

For compact notation we defined scaled variables $A'_p = A_p/(1+a_p)$ and $\tau'_p = \tau_p/(1+a_p)$.

Stacking the received vector $\mathbf{z}$, data vector $\mathbf{s}$, and noise vector $\mathbf{v}$ across subcarriers, we obtain the following input-output relationship:

$$\mathbf{z} = \mathbf{H}\mathbf{s} + \mathbf{v}. \tag{10}$$

The matrix $\mathbf{H}$ can be decomposed into a part contributed to the delays τ'_p and the Doppler rates a_p,

$$\mathbf{H} = \sum_{p=1}^{N_p} \xi_p \mathbf{\Lambda}(\tau'_p)\mathbf{\Gamma}(a_p), \tag{11}$$

where the complex path gain of the pth path is

$$\xi_p = A'_p e^{-j2\pi(f_c+\epsilon)\tau'_p}, \tag{12}$$

the matrix $\mathbf{\Gamma}(a_p)$ contains the ICI pattern $\varrho_{m,k}(a_p)$ and $\mathbf{\Lambda}_p$ is a diagonal matrix with

$$\left[\mathbf{\Lambda}(\tau'_p)\right]_{m,m} = e^{-j2\pi\frac{m}{T}\tau'_p}. \tag{13}$$

We now express the observation vector $\mathbf{z}$ as a linear combination of the multipath components

$$\mathbf{z} = \sum_{p=1}^{N_p} \xi_p \mathbf{\Lambda}(\tau'_p)\left[\mathbf{\Gamma}(a_p)\mathbf{s}\right] + \mathbf{v} = \sum_{p=1}^{N_p} \xi_p \mathbf{S}(a_p)\mathbf{w}(\tau'_p) + \mathbf{v}, \tag{14}$$

and as $\mathbf{\Lambda}(\tau'_p)\left[\mathbf{\Gamma}(a_p)\mathbf{s}\right]$ is basically an element-wise product between two vectors, we can switch the order by using the diagonal elements of $\mathbf{\Lambda}(\tau'_p)$ as a new vector $\mathbf{w}(\tau'_p)$ and the vector $\mathbf{\Gamma}(a_p)\mathbf{s}$ as a new diagonal matrix $\mathbf{S}(a_p)$. The formulation in (14) clearly shows the contribution from each discrete path with delay τ'_p and Doppler scale a_p towards the channel mixing matrix that defines the ICI pattern. We consider the following two receivers.

- **ICI-ignorant receiver** [12]: when all the paths have similar Doppler scales, proper choices of resampling factor and ϵ can render $\mathbf{H}$ close to diagonal. In this case, $\mathbf{\Gamma}(a_p)$ is approximated by an identity matrix.
- **ICI-aware receiver** [3]: we explicitly consider the possible Doppler scales for each discrete path in this case. To reduce the complexity of computing the dictionary set with a large size, we choose to retain only D off-diagonals on the templates $\mathbf{\Gamma}(a_p)$. Hence, matrix $\mathbf{H}$ is a banded mixing matrix. This means that only ICI form D directly neighoring subcarriers on each side are considered. When $D = 0$, it reduces to the ICI-ignorant receiver.

B. Multiple transmitter systems

We only consider the ICI-ignorant receiver for MIMO systems. Spatial multiplexing with N_t transmitters and N_r receivers is used, and $\mathbf{s}_\mu$ is the signal transmitted from the μth transmitter. The received signal at νth receiver is given by

$$\mathbf{z}_\nu = \sum_{\mu=1}^{N_t} \left[\sum_{p=1}^{N_p} \xi_p^{\nu,\mu} \mathbf{S}_\mu \mathbf{w}(\tau'_p) \right] + \mathbf{v}_\nu, \tag{15}$$

where $\mathbf{S}_\mu$ is a diagonal matrix with the elements of vector $\mathbf{s}_\mu$ on its main diagonal (as $\mathbf{\Gamma}(a_p)$ is an identity matrix), $\xi_p^{\nu,\mu}$ is the complex path gain of the pth path from the μth transmitter to the νth receiver and $\mathbf{v}_\nu$ is the equivalent noise, which includes both the ambient noise and the residual ICI.

III. Dictionary Construction

A. Dictionary in ICI-aware SIMO systems

To create a dictionary for sparse channel estimation, we decouple the delay τ_p from the Doppler scale a_q using different subscripts p and q. We assume that the τ'_p and a_q are from a grid of size $N_\tau \times N_a$, as

$$\tau'_p \in \left\{0, \frac{T}{\lambda K}, \frac{2T}{\lambda K}, \cdots, \frac{(N_\tau - 1)T}{\lambda K}\right\}, \quad (16)$$

$$a_q \in \{-a_{\max}, -a_{\max} + \Delta a, \cdots, a_{\max}\}, \quad (17)$$

where λ is the oversampling factor, $(N_\tau T)/(\lambda K)$ needs to be larger than the channel delay spread, and $2a_{\max}$ corresponds to the Doppler spread. Since we consider a two-dimensional grid, the dictionary size will be proportional to the product of delay and Doppler spread.

The path gain associated with a delay/Doppler tuple (τ_p, a_q) is denoted as $\xi_{p,q}$. Collecting all path gains of the same Doppler a_q into a vector $\boldsymbol{\xi}_q$, we now rewrite (14) as:

$$\mathbf{z} = \sum_{q=1}^{N_a} \mathbf{S}(a_q) \left(\sum_{p=1}^{N_\tau} \mathbf{w}(\tau'_p)\xi_{p,q} \right) + \mathbf{v} = \sum_{q=1}^{N_a} \mathbf{S}(a_q)\mathbf{W}\boldsymbol{\xi}_q + \mathbf{v}, \quad (18)$$

where the vectors $\mathbf{w}(\tau'_p)$ form a partial DFT matrix $\mathbf{W}$. Finally, we have

$$\mathbf{z} = \mathbf{A}\mathbf{x} + \mathbf{v} \quad (19)$$

with $\mathbf{x} = [\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{N_a}]^T$ and

$$\mathbf{A} = \begin{bmatrix} \mathbf{S}(a_1) & \cdots & \mathbf{S}(a_{N_a}) \end{bmatrix} (\mathbf{I}_{N_a} \otimes \mathbf{W}), \quad (20)$$

where $\otimes$ is the Kronecker product and $\mathbf{I}_n$ the identity matrix of size n. The multiplication by $\mathbf{A}$ or $\mathbf{A}^T$ can be implemented efficiently via N_a FFTs.

B. Dictionary in ICI-ignorant MIMO systems

We focus only on the ICI-ignorant receiver for the MIMO systems. Hence, only the delay dictionary is needed, which is given in (16). Instead an additional dimension will be defined by the transmitter index. Similar to before, c.f. (18), we group all path gains of various delays corresponding to the μth transmitter into a vector $\boldsymbol{\xi}^{\nu,\mu}$ and rewrite the received signal for the νth receiver as:

$$\mathbf{z}_\nu = \sum_{\mu=1}^{N_t} \mathbf{S}_\mu \left(\sum_{p=1}^{N_p} \mathbf{w}(\tau'_p)\xi_p^{\nu,\mu} \right) + \mathbf{v}_\nu = \sum_{\mu=1}^{N_t} \mathbf{S}_\mu \mathbf{W}\boldsymbol{\xi}^{\nu,\mu} + \mathbf{v}_\nu \quad (21)$$

For a more compact notation, we define the stacked vector $\mathbf{x}^\nu = \left[\boldsymbol{\xi}^{\nu,1}, \ldots, \boldsymbol{\xi}^{\nu,N_t}\right]^T$ and matrix

$$\boldsymbol{\Psi} = \begin{bmatrix} \mathbf{S}_1 & \mathbf{S}_2 & \cdots & \mathbf{S}_{N_t} \end{bmatrix} (\mathbf{I}_{N_t} \otimes \mathbf{W}), \quad (22)$$

which lead to $\mathbf{z}_\nu = \boldsymbol{\Psi}\mathbf{x}^\nu + \mathbf{v}_\nu$ to be used in the optimization problem formulated in Eq. (1).

IV. Algorithms Considered

In [3], we found that BP delivers the best performance among different channel estimators. Many solvers for the formulation in (1) are available, but it should be noted that i) $\mathbf{x}$ is complex, and ii) the matrix $\mathbf{A}$ is of large dimension, but $\mathbf{Ab}$ and $\mathbf{A}^T\mathbf{c}$ can be evaluated efficiently. We consider the following three BP solvers. As a comparison, we also include the OMP algorithm.

l1_ls: is an efficient interior-point algorithm using a preconditioned conjugate gradient method to compute the search direction [7]. To determine convergence we use the target relative tolerance, which is defined as the duality gap divided by the dual objective cost. The available software package only supports real variables, but the complex implementation is outlined as a straight-forward extension in [7].

SpaRSA: is a method based on iterative shrinkage/thresholding (IST) [8]. It computes the steepest descent direction on the l_2 norm in (1) and uses a very simple soft thresholding function related to the regularization term. SpaRSA can be viewed as a step-size improved IST. A software package is available online that supports several convergence criteria; in this paper, we use the relative step size between two consecutive estimates.

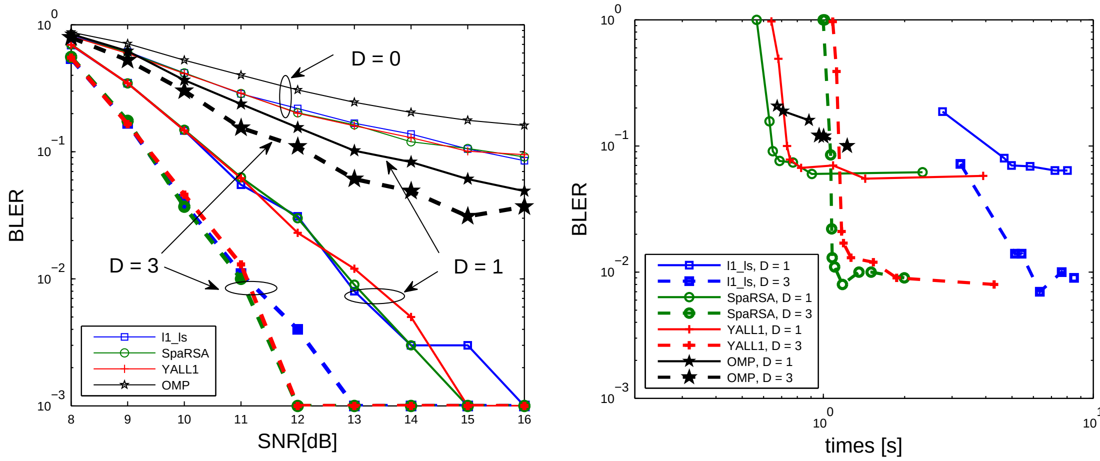
YALL1: is derived from the Alternating Direction Method (ADM), which minimizes augmented Lagrangian functions through an alternating minimization scheme and updates the multipliers after each sweep [9]. The YALL1 solver can be applied to eight L1-minimization models, hence a one-for-eight algorithm. A matlab software package is available online.

OMP: OMP is a greedy algorithm to solve the sparse reconstruction problem. The OMP algorithm selects one most possible atom at each step and updates the residual error based on all the chosen atoms [1], [13]. We use both the residual error and change in residual error as convergence criteria.

V. Numerical Results

We will use both simulation and experimental data to compare the three BP solvers: a) for a SIMO system with ICI-ignorant and ICI-aware receivers, and b) for a MIMO system with an ICI-ignorant receiver. For the experimental results, we use data recorded during the SPACE08 experiment, which was conducted off the coast of Martha's Vineyard, MA, from Oct. 14 to Nov. 1, 2008. All the numerical results were carried out under MATLAB 2008b, on a personal computer with an Intel(R) Core(TM)2 CPU 6600 @2.4 GHz processer and 3 GB of memory. Our performance metric is the BLER and the complexity metric is the average CPU time. The average CPU time is divided into two parts:

- Time Ini: the CPU time to prepare the dictionary. In practice, the dictionary can be saved in advance.
- Time Exe: the execution time, which includes the CPU time to run the optimization algorithm and the time to construct the channel frequency response matrix $\mathbf{H}$, after obtaining the time domain sparse solution.



(a) BLER vs. SNR (b) Simulation, SNR = 11 dB

Fig. 1. SIMO with ICI-aware and ICI-ignorant receivers, 16 QAM. The overview of CPU time is shown in Table I.

TABLE I

OVERVIEW OF AVERAGE CPU TIME OVER SNR FROM 8 DB TO 16 DB, SIMO CASE, SIMULATION RESULTS. TOLERANCE FOR L1_LS, SPARSA AND YALL1 ARE: 1E-2, 2E-4 AND 1E-2.

type	SIMO with ICI-aware				SIMO with ICI-ignorant	
data	D = 1 (dim ($\mathbf{x}$) = 7200)		D = 3 (dim ($\mathbf{x}$) = 7200)		D = 0 (dim ($\mathbf{x}$) = 480)	
Comparison	Time Ini	Time Exe	Time Ini	Time Exe	Time Ini	Time Exe
l1_ls	0.026 s	8.180 s	0.062 s	8.599 s	0.001 s	0.230 s
SpaRSA	0.027 s	0.765 s	0.062 s	1.180 s	0.001 s	0.022 s
YALL1	0.026 s	1.115 s	0.063 s	1.556 s	0.001 s	0.037 s
OMP	0.650 s	1.554 s	0.684 s	1.971 s	0.001 s	0.018 s

A. SIMO System with ICI-aware/ICI-ignorant Receivers

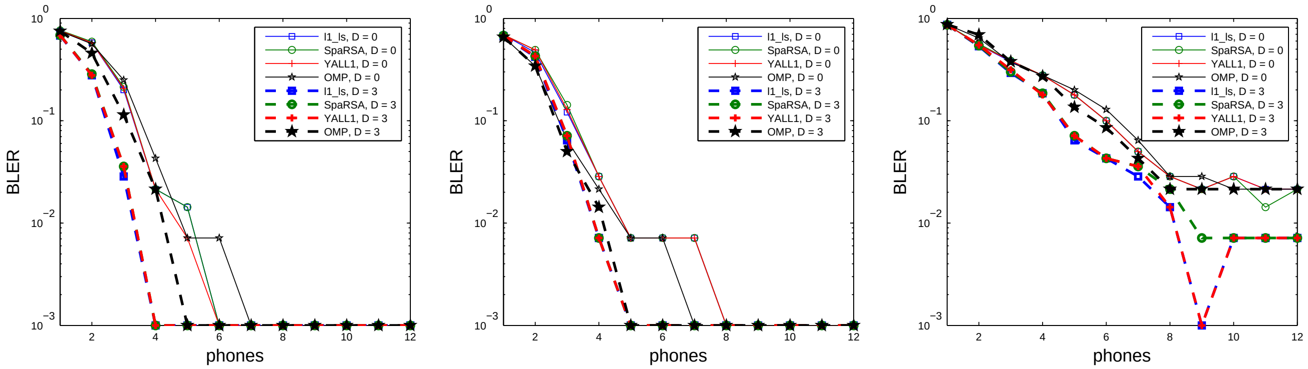
We consider the same single transmitter OFDM system as in [3] to examine the performance and complexity trade-offs; there are $K = 1024$ subcarriers, 352 of which are pilots and 96 null subcarriers, leading 576 data subcarriers. Given a bandwidth of $B = 9.77$ kHz, a symbol length of $T = 104.86$ ms, a guard interval of $T_g = 24.6$ ms, 16-QAM, and an LDPC code with rate 1/2, the overall data rate is 7.4 kb/s. After the channel has been estimated, a zero-forcing (ZF) equalizer is used, i.e., $\hat{\mathbf{s}} = \mathbf{H}^{\dagger}\mathbf{z}$, where † denote pseudo inverse. Then LDPC decoding is applied as in [14].

The simulated channel has 15 paths, with exponential distributed inter-arrival times of 1 ms average, leading to a total average delay spread of 15 ms. Each path has a separate Doppler rate, which is drawn from a uniform distribution with standard deviation of $\sigma_v = 0.2$ m/s. All channel estimators need to cover a channel delay spread equivalent to the guard interval, $T_g B = 240$, in baseband samples.

1) Simulation Results: We use a delay resolution with $\lambda = 2$, leading to $N_\tau = 480$, as well as $N_a = 15$ Doppler values totaling $N_\tau \times N_a = 7200$ unknowns, c.f. Table I. The simulation results are plotted in Fig. 1(a), where the minor differences between BP solvers can be attributed to independent noise realizations between runs. OMP performs significantly worse on severely Doppler spread channels, c.f. [3]. The average run-times are in Table I; as the run-time differences depend on the convergence criteria, we fix the SNR at 11 dB and vary the convergence threshold. We see in Fig. 1(b), that the run-times can be reduced by relaxing the convergence criterion, but at some point there is a threshold effect and the performance deteriorates.

For the ICI-ignorant receiver, the delay resolution is the same, $N_\tau = 480$, but $N_a = 1$, as no Doppler spread is assumed, leading to 480 unknowns. The simulation results are plotted in Fig. 1(a), where the ICI-ignorant receivers perform significantly worse than the ICI-aware receivers on severely Doppler spread channels, as can be expected. The run-times in Table I show that "Time Ini" is independent of the BP solvers, but increase with D. Similarly a constant increase can be seen for all BP solvers in "Time Exe" as creating the matrix $\mathbf{H}$ takes more time for larger D. For OMP we additionally need to create the matrix $\mathbf{A}^T\mathbf{A}$, leading to a higher "Time Ini". Overall the average run-times show that SpaRSA and YALL1 have similar complexity as OMP, while l1_ls's complexity is about one order higher than the others.

2) Experimental Results: As an example that highlights the Doppler spread channel model, we consider data recorded on Julian date 300 that was characterized by stormy sea conditions. Each receiver array has twelve hydrophones. Fig. 2 shows the BLER performance when combining an increasing number of phones for receivers S1 (60 m), S3 (200 m) and S5 (1,000 m from the transmitter) that generally follows the



(a) S1 (60 m) (b) S3 (200 m) (c) S5 (1000 m)

Fig. 2. SIMO with ICI-ignorant and ICI-aware receiver, 16-QAM, Julian date 300, SPACE08.

TABLE II
OVERVIEW OF CPU TIME, SIMO CASE, EXPERIMENTAL RESULTS, SPACE08 JULIAN DATE 300, S5. TOLERANCE FOR L1_LS, SPARSA AND YALL1 ARE: 1E-2, 2E-4 AND 1E-2.

type	SIMO with ICI-aware				SIMO with ICI-ignorant	
data	D = 1 (dim ($\mathbf{x}$) = 7200)		D = 3 (dim ($\mathbf{x}$) = 7200)		D = 0 (dim ($\mathbf{x}$) = 480)	
Comparison	Time Ini	Time Exe	Time Ini	Time Exe	Time Ini	Time Exe
l1_ls	0.026 s	7.371 s	0.061 s	7.804 s	5e-5 s	0.261 s
SpaRSA	0.026 s	0.957 s	0.061 s	1.367 s	5e-5 s	0.029 s
YALL1	0.026 s	1.175 s	0.061 s	1.616 s	5e-5 s	0.040 s
OMP	0.639 s	3.597 s	0.697 s	4.663 s	3e-4 s	0.046 s

same trends as the simulated data. The run-times of S5 are shown in Table II, which largely match the simulation results. The only obvious difference is that OMP seems to need more computation time with the experimental data. Since the OMP run time is proportional to the number of active dictionary columns, it seems that our simulation with fifteen paths was underestimating the number of significant paths.

B. *MIMO with ICI-ignorant Receiver*

The multiple-input multiple-output (MIMO) OFDM system is explained in detail in [10]. Specifically there are $K = 1024$ subcarriers, 256 of which are pilots and 96 null subcarriers, leading 672 data subcarriers. Due to fewer pilots and independent data streams on each transmitter the data rate is higher, specifically for $N_t = 2$ the data rate is 20.8 kb/s. As each transmitter leads to an independent channel realization, the number of unknowns scales with the number of transmitters N_t. Channel estimation is performed separately per receiver. After the channel has been estimated, we use the MIMO detector of [10], which consists of a hybrid use of successive interference cancellation and soft MMSE equalizer with a priori information to perform data detection on each data subcarrier.

1) Simulation Results: The delay resolution is as previously, $\lambda = 2$, but operating ICI-ignorant, $N_a = 1$, leading to $N_\tau \times N_a \times N_t = 960$ unknowns. The simulated channels for each transmitter-receiver pair are generated as for the SIMO case, but with no Doppler effects, $\sigma_v = 0$ m/s. The BLER performance is shown in Fig. 3(a), again all BP solvers perform equally well. The run-times (see Table III) are shorter

TABLE III
OVERVIEW OF CPU TIME, MIMO WITH $N_t = 2$ AND $N_r = 4$. SIMULATION AND EXPERIMENTAL RESULTS. TOLERANCE FOR L1_LS, SPARSA AND YALL1 ARE: 1E-2, 2E-4 AND 1E-2.

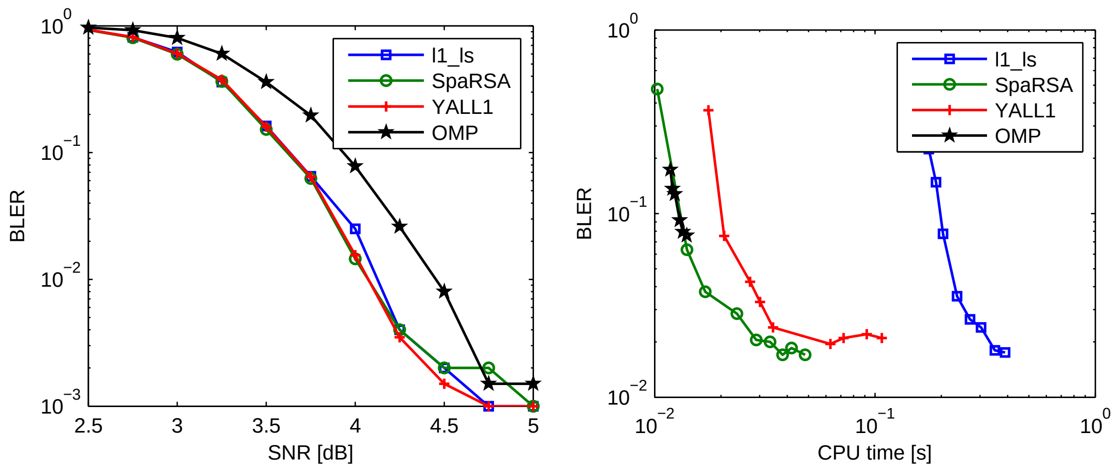
type	MIMO with ICI-ignorant (D = 0 (dim ($\mathbf{x}$) = 960))			
data	Simulation		Experimental	
Comparison	Time Ini	Time Exe	Time Ini	Time Exe
l1_ls	0.001 s	0.388 s	0.001 s	0.356 s
SpaRSA	0.001 s	0.035 s	0.001 s	0.033 s
YALL1	0.001 s	0.059 s	0.001 s	0.072 s
OMP	0.053 s	0.016 s	0.054 s	0.056 s

by about a factor of twenty against the SIMO counterparts, due to the smaller problem dimension. The convergence study shows a softer behavior in Fig. 3(b) than that previously reported in Fig. 1(b). The performance stays stable over a large range of values.

2) Experimental Results: As the MIMO receiver operates in an ICI-ignorant fashion, we focus on Julian date 298, which has calm sea state. The performance for receivers S1, S3, and S5 is shown in Fig. 4, which again follows closely what we observed in simulation. The run-times in Table III show again an increased OMP complexity.

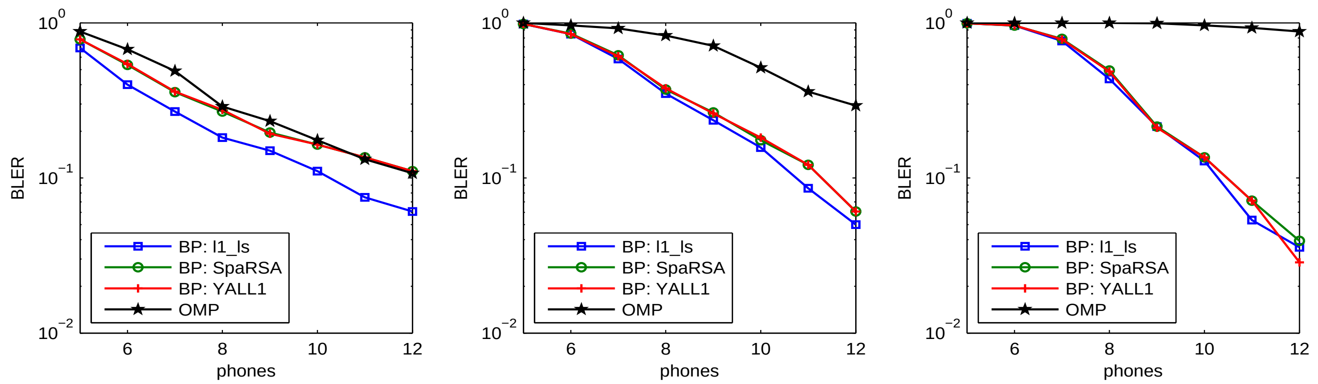
VI. CONCLUSION

In this paper we compared three BP solvers, l1_ls, SpaRSA, and YALL1 for sparse channel estimation in underwater acoustic OFDM systems. Based on simulation and experimental results, we found that SpaRSA and YALL1 reduce run-times by about one order of magnitude in comparison to l1_ls, at



(a) Simulation (b) Simulation, SNR = 4 dB

Fig. 3. MIMO with ICI-ignorant receivers, $N_t = 2$, $N_r = 4$, 16QAM, 1000 runs.



(a) S1 (60 m) (b) S3 (200 m) (c) S5 (1000 m)

Fig. 4. MIMO with ICI-ignorant receivers, SPACE08, Julian date 298.

the same block-error-rate performance. That brings their run-times to values comparable to OMP, which cannot match their block-error-rate performance. Hence, SpaRSA and YALL1 are appealing candidates to be included in real-time underwater OFDM modem development.

Acknowledgement

We are grateful to Prof. Yin Zhang from Rice University for providing the early versions of the YALL1 algorithm [9] and his comments during the early development of this paper.

References

[1] W. Li and J. C. Preisig, "Estimation of rapidly time-varying sparse channels," *IEEE J. Ocean. Eng.*, vol. 32, no. 4, pp. 927–939, Oct. 2007.
[2] T. Kang and R. A. Iltis, "Iterative carrier frequency offset and channel estimation for underwater acoustic OFDM systems," *IEEE J. Select. Areas Commun.*, vol. 26, no. 9, pp. 1650–1661, Dec. 2008.
[3] C. R. Berger, S. Zhou, J. Preisig, and P. Willett, "Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing," Mar. 2010.
[4] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inform. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.
[5] D. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[6] R. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, Jul. 2007.
[7] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale l_1-regularized least squares," *IEEE J. Select. Topics Signal Proc.*, vol. 1, no. 4, pp. 606–617, Dec. 2007.
[8] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Signal Processing*, vol. 57, no. 7, pp. 2479–2493, Jul. 2009.
[9] Y. Zhang, "User's guide for yall1: Your algorithms for l1 optimization," downloable at http://www.caam.rice.edu/~optimization/L1/YALL1/.
[10] B. Li, J. Huang, S. Zhou, K. Ball, M. Stojanovic, L. Freitag, and P. Willett, "MIMO-OFDM for high rate underwater acoustic communications," *IEEE J. Ocean. Eng.*, vol. 34, no. 4, pp. 634–644, Oct. 2009.
[11] H. Yan, S. Zhou, Z. Shi, and B. Li, "A DSP implementation of OFDM acoustic modem," in *Proc. of the ACM Intl. Workshop on Underwater Networks (WUWNet)*, Montréal, Québec, Canada, Sep. 2007.
[12] B. Li, S. Zhou, M. Stojanovic, L. Freitag, and P. Willett, "Multicarrier communication over underwater acoustic channels with nonuniform Doppler shifts," *IEEE J. Ocean. Eng.*, Apr. 2008.
[13] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inform. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.
[14] J. Huang, S. Zhou, and P. Willett, "Nonbinary LDPC coding for multicarrier underwater acoustic communication," *IEEE J. Select. Areas Commun.*, vol. 26, no. 9, pp. 1684–1696, Dec. 2008.