Signal Processing 101 (2014) 174–191

Contents lists available at ScienceDirect

Signal Processing

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

Sparsity-aware multi-source RSS localization ☆

Hadi Jamali-Rad *, Hamid Ramezani, Geert Leus

Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2826 CD Delft, The Netherlands

ARTICLE INFO

Article history:
Received 18 October 2013
Received in revised form
19 January 2014
Accepted 13 February 2014
Available online 22 February 2014

Keywords:
Multi-source localization
Multipath environments
RSS fingerprinting
Sparse reconstruction

ABSTRACT

We tackle the problem of localizing *multiple sources* in multipath environments using received signal strength (RSS) measurements. The existing *sparsity-aware* fingerprinting approaches only use the RSS measurements (autocorrelations) at different access points (APs) separately and ignore the potential information present in the cross-correlations of the received signals. We propose to reformulate this problem to exploit this information by introducing a novel fingerprinting paradigm which leads to a significant gain in terms of number of identifiable sources. Besides, we further enhance this newly proposed approach by incorporating the information present in the other time lags of the autocorrelation and cross-correlation functions. An interesting by-product of the proposed approaches is that under some conditions we can convert the given under-determined problem to an overdetermined one and efficiently solve it using classical least squares (LS). Moreover, we also approach the problem from a frequency-domain perspective and propose a method which is *blind* to the statistics of the source signals. Finally, we incorporate the so-called concept of finite-alphabet sparsity in our framework for the case where the sources have a similar power. Our extensive simulation results illustrate a good performance as well as a significant detection gain for the introduced multi-source RSS fingerprinting methods.

1. Introduction

Precise localization of multiple sources is a fundamental problem which has received a lot of attention recently [1]. Many different approaches have been proposed in the literature to recover the location of the sources based on time-of-flight (ToF), time-difference-of-arrival (TDOA) or received-signal-strength (RSS) measurements. A traditional wisdom in RSS-based localization tries to extract distance information from the RSS measurements. However, this approach fails to provide accurate location estimates due to the complexity and unpredictability of the wireless channel. This has motivated another category of RSS-based positioning, the so-called location fingerprinting, which discretizes the physical space into grid points (GPs) and creates a map representing the space by assigning to every GP a location-dependent RSS parameter, one for every access point (AP). The location of the source is then estimated by comparing real-time measurements with the fingerprinting map at the source or APs, for instance using K-nearest neighbors (KNN) [2] or Bayesian classification (BC) [3].

A deeper look into the grid-based fingerprinting localization problem reveals that the source location is unique in the spatial domain, and can thus be represented by a 1-sparse vector. This motivated the use of compressive sampling (CS) [4] to

☆ This work was supported by NWO-STW under the VICI program (Project 10382). Part of this work has been presented at the European Signal Processing Conference (EUSIPCO'13), Marrakech, September 2013.

* Corresponding author. Tel.: +31 152786280; fax: +31 152786190.
E-mail addresses: h.jamalirad@tudelft.nl (H. Jamali-Rad), h.mashhadiramezani@tudelft.nl (H. Ramezani), g.j.t.leus@tudelft.nl (G. Leus).

http://dx.doi.org/10.1016/j.sigpro.2014.02.010
0165-1684

recover the location of the source using a few measurements by solving an ℓ_1-norm minimization problem. This idea illustrated promising results for the first time in [5,6] as well as in the following works [7–13]. In [7–9], a two-step CS-based indoor localization algorithm for multiple targets is proposed. In the first coarse localization step, the idea of cluster matching is used to determine in which cluster the targets are located. This is followed by a fine localization step in which CS is used to recover sparse signals from a small number of noisy measurements. In [10,11] it is proposed to use a joint distributed CS (JDCS) method in a practical localization scenario in order to exploit the common sparse structure of the received measurements to localize *one* target. Further, for a similar localization scenario as [10,11], in [12] the encryption capability of CS is demonstrated as CS shows robustness to potential intrusions of unauthorized entities. In [13], finally, a greedy matching pursuit algorithm is proposed for RSS-based target counting and localization with high accuracy.

Although our focus is on RSS-based source localization in this paper, let us also shortly review some existing sparsity-aware studies in the TDOA domain. Interestingly, not much work can be found on TDOA-based source localization within a sparse representation framework. In [14], single-source TDOA-based localization is proposed wherein the sparsity of the multipath channel is exploited for time-delay estimation. On the other hand, in [15], the source sparsity is exploited to simplify the hyperbolic source localization problem into an ℓ_1-norm minimization. However, the algorithm in [15] treats different sources separately, i.e., it is in principle a single-source localization approach. In [16], we have investigated the problem of sparsity-aware passive localization of multiple sources from TDOA measurements.

Coming back to RSS-based sparsity-aware localization, existing algorithms only make use of the signal/RSS readings at the different receivers (or APs), separately. However, there is potential information in the cross-correlations of these received signals at the different APs, which has not yet been exploited. In [17], we have proposed to reformulate the sparse localization problem within a *single-path* channel environment so that we can make use of the cross-correlations of the signal readings at the different APs. In this paper, we extend our basic idea in [17] by presenting the following main contributions:

I. First of all, in contrast to [17], we consider a realistic multipath channel model (simulated by a *room impulse response* (RIR) generator [18]), and we show that our idea can also be employed in a realistic multipath environment.
II. Second, we analytically show that this new framework can provide a considerable amount of extra information compared to classical algorithms which leads to a significant improvement in terms of the number of identifiable sources as well as localization accuracy. To guarantee a high quality reconstruction, we also numerically assess the restricted isometry property (RIP) of our proposed fingerprinting maps.
III. In order to further improve the potential of our novel framework in terms of number of identifiable sources, we also propose to exploit extra information in the time domain. Particularly, the information in other lags than the zeroth lag of the autocorrelation and cross-correlation functions can be exploited to construct a larger fingerprinting map.
IV. We propose a novel idea to deal with the cases where there is no knowledge about the statistics of the transmitted signals by the source nodes (SNs). This basically makes it possible to perform fingerprinting in a blind fashion with respect to (w.r.t.) the statistics of the transmitted signals. This blind approach is mainly based on a proper filter bank design to approach the fingerprinting problem from the frequency domain. Moreover, we also show that incorporating this information in the frequency domain improves the performance in terms of number of identifiable sources compared to the original proposed approach.
V. We show that if the sources transmit the same signal power the sparse vector of interest will contain finite-alphabet elements. In such cases, we propose to recover the locations by taking the finite-alphabet property of the non-zero elements of the sparse vector into account, which we refer to as finite-alphabet sparsity. We show that including this information leads to a considerable reconstruction gain.

Note that the sources considered here are *non-cooperative*, i.e., the sources do not emit radio signals with the purpose of localization, but the signals are intended for communications and we exploit them for localization. The proposed algorithms can be applied in indoor or outdoor environments. For instance, monitoring non-cooperative sources broadcasting CDMA signals can be an example of our application domain. However, there is no limitation to employ the proposed ideas in wireless LAN (WLAN) or wireless sensor networks (WSNs) operating in a centralized fashion with a wired backbone.

The rest of the paper is organized as follows. In Section 2, the signal and network model under consideration are explained. In Section 3, the classical RSS-based fingerprinting localization as well as our proposed fingerprinting idea are explained. The RIP of the proposed fingerprinting maps is also numerically assessed in this section. Section 4 explains how extra information in the time domain can be exploited to further enhance the performance of our proposed fingerprinting idea. The idea of blind fingerprinting using frequency domain information is presented in Section 5. Section 6 explains the idea of using finite-alphabet sparsity. Extensive simulation results in Section 7 corroborate our analytical claims in several scenarios. Finally in Section 8, after a short discussion on computational complexity of the proposed algorithms, the paper is wrapped up with brief concluding remarks.

2. Problem definition

Consider that we have M access points (APs) distributed over an area which is discretized into N cells each represented by its central grid point (GP). The APs can be located anywhere. We consider K *non-cooperative* source nodes (SNs) which are

randomly located either on these GPs (*on-grid* scenario) or anywhere (*off-grid* scenario). We assume that the APs are connected to each other by a wired backbone so that they can cooperate by exchanging their signal readings. We also assume that the APs are synchronized, which is feasible especially considering the wired backbone. Now, if the k-th SN broadcasts a time domain signal $s_k(t)$, the received signal at the m-th AP can be expressed by

$$x_{m,k}(t) = \sum_{l=1}^{L} h_{l,m,k} s_k(t-\tau_{l,m,k}) + n_m(t), \tag{1}$$

where we consider an L-path channel with $h_{l,m,k}$ and $\tau_{l,m,k}$ respectively denoting the channel coefficient and time-delay of the l-th path from the k-th SN to the m-th AP; $n_m(t)$ is the additive noise at the m-th AP. Our assumptions on the signal and noise models are as follows:

A.1 The signals $s_k(t)$ are assumed to be ergodic, mutually uncorrelated sequences, i.e., $\mathbb{E}\{s_k(t)s_{k'}^*(t')\} = \eta_k r_k(t-t')\delta_{k-k'}$, with η_k being the k-th signal power, $r_k(\tau)$ the normalized signal correlation function with $r_k(0)=1$, and δ_k the unit impulse function. Meanwhile, $\mathbb{E}\{\cdot\}$ denotes the statistical expectation which is equal to temporal averaging due to the ergodic property of the signals.

A.2 The noises $n_m(t)$ are assumed to be ergodic, mutually uncorrelated white sequences, i.e., $\mathbb{E}\{n_m(t)n_{m'}^*(t')\} = \sigma_n^2\delta(t-t')\delta_{m-m'}$, with $\sigma_n^2 = N_0 B$ the variance of the additive noise with density N_0 within the operating frequency bandwidth B, and $\delta(t)$ the Dirac impulse function.

A.3 The transmitted signals are uncorrelated with the additive noise, i.e., $\mathbb{E}\{s_k(t)n_m(t')\} = 0,\ \forall t, t'$ and $\forall m, k$.

A.4 Throughout the paper we consider $r_k(\tau) = r(\tau), \forall k$ and we assume it to be known a priori or acquired through training, unless otherwise mentioned. Note that under this assumption the SNs can still be considered *non-cooperative* in the sense that they do not cooperate by exchanging information.

A.5 As a more general case, we sometimes also consider $r_k(\tau) \neq r_{k'}(\tau)$ and assume they are unknown. This requires an approach which is *blind* to the $r_k(\tau)$'s.

From (1), the total received signal at the m-th AP can be written as

$$x_m(t) = \sum_{k=1}^{K} x_{m,k}(t) = \sum_{k=1}^{K}\sum_{l=1}^{L} h_{l,m,k} s_k(t-\tau_{l,m,k}) + n_m(t). \tag{2}$$

It is worth pointing out that in a general sense, the problem under consideration is a *passive* localization problem as $x_m(t)$ cannot be decomposed into its components $x_{m,k}(t)$. The problem here is to use the total received signals at the APs to localize the SNs simultaneously. In the following, we propose a novel RSS-based fingerprinting paradigm to localize the SNs within a multipath environment.

3. Sparsity-aware RSS localization

Localizing multiple SNs using their received signals is a non-trivial problem which can be converted into a linear problem by taking into account the sparsity of the SNs in the spatial domain. In order to be able to incorporate the sparsity, we define a grid structure in space consisting of N GPs. Next, we perform localization in two phases; first, we construct the fingerprinting map in an initialization phase by either training or if possible analytical computation. More specifically, if training is considered, a training SN (transmitting $s_0(t)$ with signal correlation $r(\tau)$ and power $\eta_0 = 1$) is put on every GP, one after the other, and the signal readings at all the APs are used to construct the map. Alternatively, the channel coefficients and the time-delays of the received signals at all the APs can be computed analytically (e.g., using the RIR generator [18]) whereas the statistics of the $s_k(t)$'s, i.e., the $r(\tau)$, are assumed to be known (or measured) beforehand. Notably, an important advantage of analytically computing the map is avoiding an exhaustive training procedure. In the second phase, the so-called *run-time phase*, real-time multi-source measurements of the sources with similar statistics as in the initialization phase are collected and processed to recover the locations of the SNs.

It is also worth highlighting that the case of *off-grid* source localization can for instance be handled using adaptive mesh refinement algorithms as explained in [19] or by finding the "grid mismatch" using sparse total least squares (STLS) ideas as we proposed in [16], but this is left as future work due to space limitations. In this paper, we confine ourselves to finding the closest GPs to the off-grid sources as explained in Section 7.3.

3.1. Classical sparsity-aware RSS localization (SRL)

One way to compute the RSS is by taking the zeroth lag of the autocorrelation function of the received time-domain signals at the APs as

$$\begin{aligned} y_m &= \mathbb{E}\{x_m(t)x_m^*(t)\} \\ &= \mathbb{E}\left\{\left(\sum_{k=1}^{K}\sum_{l=1}^{L} h_{l,m,k} s_k(t-\tau_{l,m,k}) + n_m(t)\right)\right. \end{aligned}$$

$$\times\left(\sum_{k'=1}^{K}\sum_{l'=1}^{L} h^*_{l',m,k'}s^*_{k'}(t-\tau_{l',m,k'})+n^*_m(t)\right)\Bigg\}$$

$$=\mathbb{E}\left\{\sum_{k=k'=1}^{K}\sum_{l=1}^{L}\sum_{l'=1}^{L} h_{l,m,k}h^*_{l',m,k}s_k(t-\tau_{l,m,k})s^*_k(t-\tau_{l',m,k})\right\}+\mathbb{E}\{n_m(t)n^*_m(t)\}$$

$$=\sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{l'=1}^{L} h_{l,m,k}h^*_{l',m,k}r(\tau_{l',m,k}-\tau_{l,m,k})\eta_k+\sigma_n^2, \tag{3}$$

which for a single-path channel model boils down to $y_m=\sum_{k=1}^{K}|h_{m,k}|^2\eta_k+\sigma_n^2$. Notably, the third equality follows from A.1 and A.3 and the last equality follows from A.2 and A.4, as detailed in Section 2. Interestingly, if we ignore the effect of the noise for the time being, the last expression in (3) shows that the RSS at AP_m is a summation of K location-dependent (through delays and channel coefficients) terms $\sum_{l=1}^{L}\sum_{l'=1}^{L}h_{l,m,k}h^*_{l',m,k}r(\tau_{l',m,k}-\tau_{l,m,k})$. This means that if these K components could be recognized, the locations can be estimated from them, which motivates choosing them as *fingerprints* of the sources. Now, in order to be able to do this, we consider that the SNs can only be located on a finite set of positions determined by N GPs. Therefore, if we measure/compute the fingerprints of the N GPs, the corresponding N fingerprints can be stacked in a vector expressed by

$$\boldsymbol{\psi}_m=\left[\sum_{l=1}^{L}\sum_{l'=1}^{L} h^g_{l,m,1}h^{g*}_{l',m,1}r(\tau^g_{l',m,1}-\tau^g_{l,m,1}),\ldots,\sum_{l=1}^{L}\sum_{l'=1}^{L} h^g_{l,m,N}h^{g*}_{l',m,N}r(\tau^g_{l',m,N}-\tau^g_{l,m,N})\right]^T, \tag{4}$$

where $(\cdot)^g$ denotes values being measured/computed for the GPs. Thus, using (4), (3) can be rewritten for a grid structure as

$$y_m=\boldsymbol{\psi}_m^T\boldsymbol{\theta}+\sigma_n^2, \tag{5}$$

where $\boldsymbol{\theta}$ is an $N\times 1$ vector containing all zeros except for K non-zero elements with indices related to the locations of the K sources and values equal to the η_k's. The same holds for the other APs with the same $\boldsymbol{\theta}$, which helps us to stack the y_m's and $\boldsymbol{\psi}_m$'s for different APs as

$$\mathbf{y}=\boldsymbol{\Psi}\boldsymbol{\theta}+\mathbf{p}_n, \tag{6}$$

where $\mathbf{p}_n=\sigma_n^2\mathbf{1}_M$ with $\mathbf{1}_M$ the $M\times 1$ vector of all ones, $\mathbf{y}=[y_1,\ldots,y_M]^T$ and $\boldsymbol{\Psi}=[\boldsymbol{\psi}_1,\ldots,\boldsymbol{\psi}_M]^T$. Defining $\mathbf{x}(t)=[x_1(t),\ldots,x_M(t)]^T$, it is clear that

$$\mathbf{y}=\mathbb{E}\{\mathbf{x}(t)\odot\mathbf{x}^*(t)\}, \tag{7}$$

where $\odot$ denotes the element-wise Hadamard product. As is clear from (6), $\mathbf{y}$ is the K-sparse RSS characterized by the fingerprinting map $\boldsymbol{\Psi}$ as given by

$$\boldsymbol{\Psi}^T=\sum_{l=1}^{L}\sum_{l'=1}^{L}\left\{\begin{bmatrix} h^g_{l,1,1}h^{g*}_{l',1,1}r(\tau^g_{l',1,1}-\tau^g_{l,1,1}) & \cdots & h^g_{l,M,1}h^{g*}_{l',M,1}r(\tau^g_{l',M,1}-\tau^g_{l,M,1})\\ h^g_{l,1,2}h^{g*}_{l',1,2}r(\tau^g_{l',1,2}-\tau^g_{l,1,2}) & \cdots & h^g_{l,M,2}h^{g*}_{l',M,2}r(\tau^g_{l',M,2}-\tau^g_{l,M,2})\\ \vdots & \ddots & \vdots\\ h^g_{l,1,N}h^{g*}_{l',1,N}r(\tau^g_{l',1,N}-\tau^g_{l,1,N}) & \cdots & h^g_{l,M,N}h^{g*}_{l',M,N}r(\tau^g_{l',M,N}-\tau^g_{l,M,N})\end{bmatrix}\right\}. \tag{8}$$

Note that if the SNs have different signal powers, estimating $\boldsymbol{\theta}$ will also return the signal powers as a by-product. Solving (6) with classical LS produces a poor estimate due to the under-determined nature of the problem ($M<N$). Instead, sparse reconstruction techniques (or CS) aim to reconstruct $\boldsymbol{\theta}$ by taking the source sparsity concept into account. It is worth mentioning that here we have a natural compression in the problem, in the sense that the number of measurements is limited to the number of APs (M), which in many practical scenarios is much less than the number of GPs (N). Hence, using (6), $\boldsymbol{\theta}$ can be well-recovered by solving the following ℓ_1-norm minimization:

$$\hat{\boldsymbol{\theta}}_{\mathrm{SRL}}=\arg\min_{\boldsymbol{\theta}}\ \|\mathbf{y}-\boldsymbol{\Psi}\boldsymbol{\theta}\|_2^2+\lambda\|\boldsymbol{\theta}\|_1, \tag{9}$$

where λ is a regularization parameter that controls the trade-off between sparsity and reconstruction fidelity of the estimated $\boldsymbol{\theta}$. The problem (9) can efficiently be solved using several algorithms including the well-known LASSO [7]. We would like to stress that (even though modified to fit our setup) the discussed SRL represents the existing classical sparsity-aware RSS localization idea in the literature [7] and it is modified and presented here for the sake of comparison.

Remark 1 (*Identifiability of SRL*). To elaborate on the identifiability of localization using SRL, it is worth mentioning that for classical multi-source (2-dimensional) RSS-based localization, as long as there are $M\geq 3$ APs (not lying on a straight line), the SNs can be uniquely identified and localized. On the other hand, the sparse reconstruction-based nature of SRL imposes an extra constraint $M\geq 2K$ ($M\geq 3$ should also be satisfied) because for a perfect reconstruction we require every $2K$-column subset of $\boldsymbol{\Psi}$ to be full column rank so that we can reconstruct a K-sparse $\boldsymbol{\theta}$. All in all, this leads to the necessary condition $M\geq\max(2K,3)$ for identifiability and reconstruction. □

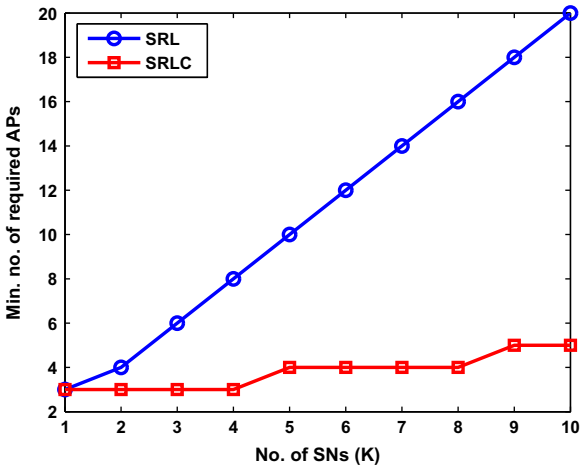

Fig. 1. Identifiability gain of SRLC compared to SRL.

3.2. Sparsity-aware RSS localization via cooperative APs (SRLC)

As explained in the previous subsection, the existing sparsity-aware RSS-based algorithms represented by the SRL, only make use of the zeroth lag of the autocorrelation function (signal strength) of the signals received at each AP separately and ignore the potential information present in the cross-correlation of this information. We propose to reformulate the problem so that we can exploit this extra information by a cooperation among the APs. This new model requires the construction of a new fingerprinting map as will be explained subsequently. Let us instead of the autocorrelations of the received signals at each AP, this time also compute the cross-correlations as

$$\begin{aligned} y_{m,m'} &= \mathbb{E}\{x_m(t)x^*_{m'}(t)\} \\ &= \mathbb{E}\left\{\left(\sum_{k=1}^{K}\sum_{l=1}^{L} h_{l,m,k}s_k(t-\tau_{l,m,k})+n_m(t)\right)\left(\sum_{k'=1}^{K}\sum_{l'=1}^{L} h^*_{l',m',k'}s^*_{k'}(t-\tau_{l',m',k'})+n^*_{m'}(t)\right)\right\} \\ &= \mathbb{E}\left\{\sum_{k=k'=1}^{K}\sum_{l=1}^{L}\sum_{l'=1}^{L} h_{l,m,k}h^*_{l',m',k}s_k(t-\tau_{l,m,k})s^*_k(t-\tau_{l',m',k})\right\}+\mathbb{E}\{n_m(t)n^*_{m'}(t)\} \\ &= \sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{l'=1}^{L} h_{l,m,k}h^*_{l',m',k}r(\tau_{l',m',k}-\tau_{l,m,k})\eta_k+\sigma_n^2\delta_{m-m'}, \end{aligned} \tag{10}$$

which for a single-path channel model boils down to $y_{m,m'}=\sum_{k=1}^{K}h_{m,k}h^*_{m',k}r(\tau_{m',k}-\tau_{m,k})\eta_k+\sigma_n^2\delta_{m-m'}$. Again, the third equality follows from A.1 and A.3 and the last equality follows from A.2 and A.4, as detailed in Section 2. Similar to the case of the SRL, if we ignore the noise effect for the time being, (10) again introduces a location-dependent fingerprint $\sum_{l=1}^{L}\sum_{l'=1}^{L}h_{l,m,k}h^*_{l',m',k}r(\tau_{l',m',k}-\tau_{l,m,k})$ for the K sources. Thus, by considering the GPs as the only possible locations of the SNs, if we measure/compute the fingerprints of the N GPs, the corresponding N fingerprints can be stacked in a vector as given by

$$\tilde{\boldsymbol{\psi}}_{m,m'}=\left[\sum_{l=1}^{L}\sum_{l'=1}^{L} h^g_{l,m,1}h^{g*}_{l',m',1}r(\tau^g_{l',m,1}-\tau^g_{l,m',1}),\ldots,\sum_{l=1}^{L}\sum_{l'=1}^{L} h^g_{l,m,N}h^{g*}_{l',m',N}r(\tau^g_{l',m,N}-\tau^g_{l,m',N})\right]^T, \tag{11}$$

and therefore using (11), (10) can be rewritten for a grid structure as

$$y_{m,m'}=\tilde{\boldsymbol{\psi}}^T_{m,m'}\boldsymbol{\theta}+\sigma_n^2\delta_{m-m'}, \tag{12}$$

where $\boldsymbol{\theta}$ is the same K-sparse vector as in the case of the SRL. In order to end up with a similar expression as (6), we can stack the M^2 different $y_{m,m'}$'s and $\tilde{\boldsymbol{\psi}}_{m,m'}$'s leading to

$$\tilde{\mathbf{y}}=\tilde{\boldsymbol{\Psi}}\boldsymbol{\theta}+\tilde{\mathbf{p}}_n, \tag{13}$$

where

$$\tilde{\mathbf{y}}=[y_{1,1},\ldots,y_{1,M},\ldots,y_{M,1},\ldots,y_{M,M}]^T, \tag{14}$$

$$\tilde{\boldsymbol{\Psi}}=[\tilde{\boldsymbol{\psi}}_{1,1},\ldots,\tilde{\boldsymbol{\psi}}_{1,M},\ldots,\tilde{\boldsymbol{\psi}}_{M,1},\ldots,\tilde{\boldsymbol{\psi}}_{M,M}]^T, \tag{15}$$

and $\tilde{\mathbf{p}}_n = \mathrm{vec}(\sigma_n^2 \mathbf{I}_M)$. Clearly, in contrast to $\mathbf{y} = \mathbb{E}\{\mathbf{x}(t) \odot \mathbf{x}^*(t)\}$, this time we compute $\tilde{\mathbf{y}} = \mathbb{E}\{\mathbf{x}(t) \otimes \mathbf{x}^*(t)\}$ where $\otimes$ represents the Kronecker product. Hence, now $\tilde{\mathbf{y}}$ is a K-sparse vector parametrized using a fingerprinting map of size $M^2 \times N$:

$$\tilde{\mathbf{\Psi}}^T = \sum_{l=1}^{L} \sum_{l'=1}^{L} \left\{ \begin{bmatrix} h_{l,1,1}^{g*} h_{l',1,1}^{g} r(\tau_{l,1,1}^{g} - \tau_{l',1,1}^{g}) & \cdots & h_{l,M,1}^{g*} h_{l',M,1}^{g} r(\tau_{l,M,1}^{g} - \tau_{l',M,1}^{g}) \\ h_{l,1,2}^{g*} h_{l',1,2}^{g} r(\tau_{l,1,2}^{g} - \tau_{l',1,2}^{g}) & \cdots & h_{l,M,2}^{g*} h_{l',M,2}^{g} r(\tau_{l,M,2}^{g} - \tau_{l',M,2}^{g}) \\ \vdots & \ddots & \vdots \\ h_{l,1,N}^{g*} h_{l',1,N}^{g} r(\tau_{l,1,N}^{g} - \tau_{l',1,N}^{g}) & \cdots & h_{l,M,N}^{g*} h_{l',M,N}^{g} r(\tau_{l,M,N}^{g} - \tau_{l',M,N}^{g}) \end{bmatrix} \right\}. \tag{16}$$

Remark 2 (*Identifiability of SRLC*). For the enhanced model, we require $M^2 \geq 2K$ and $M \geq 3$ which results in the necessary identifiability condition $M \geq \max(\sqrt{2K}, 3)$. Notably, for the special case where the channel coefficients are real, i.e., $y_{m,m'} = y_{m',m}, \forall m, m'$, we obtain only $M(M+1)/2$ different elements in $\tilde{\mathbf{y}}$ and the same number of rows in $\tilde{\mathbf{\Psi}}$. For such a case, we require $M(M+1)/2 \geq 2K$ and $M \geq 3$ which results in the necessary identifiability condition $M \geq \max(\lceil -1/2 + \sqrt{16K+1}/2 \rceil, 3)$, where $\lceil \cdot \rceil$ denotes the ceiling operator. □

As can be seen, in general, the newly proposed fingerprinting model given by (13) provides us with a set of M^2 linear equations instead of only M as in (6). This added information ($M^2 - M$ extra rows), obtained by taking cross-correlations of the received signals at the different APs into account, makes it possible for the system to localize a larger number of SNs with a fixed number of APs. This particularly becomes even more important when the physical conditions of the covered area limit the number of possible APs. By considering the statements of Remarks 1 and 2, this gain is illustrated in Fig. 1 using the minimum number of APs required to identify K SNs simultaneously. As can be seen, the proposed fingerprinting paradigm is theoretically capable of localizing the same number of SNs with much fewer APs. The new sparsity-aware localization problem in (13) can now be solved by considering the following two cases:

- *Case* I: $N > M^2$. In this case, by considering the sparse structure of $\boldsymbol{\theta}$, the extra information enables us to locate more SNs by solving the following ℓ_1-norm minimization problem (for instance using LASSO):

$$\hat{\boldsymbol{\theta}}_{\mathrm{SRLC}} = \arg\min_{\boldsymbol{\theta}} \|\tilde{\mathbf{y}} - \tilde{\mathbf{\Psi}}\boldsymbol{\theta}\|_2^2 + \lambda \|\boldsymbol{\theta}\|_1, \tag{17}$$

- *Case* II: $N \leq M^2$. Since $\tilde{\mathbf{\Psi}}$ has generally full column rank in this case, no matter what the structure of $\boldsymbol{\theta}$ might be, even if it is not sparse, it can be efficiently recovered by ordinary LS as

$$\hat{\boldsymbol{\theta}}_{\mathrm{LS}} = \tilde{\mathbf{\Psi}}^{\dagger} \tilde{\mathbf{y}}, \tag{18}$$

where $(\cdot)^{\dagger}$ represents the pseudo-inverse.

It is worth pointing out that the idea proposed in this subsection can be further improved by exploiting extra information from the time and frequency domains. This basically motivates Sections 4 and 5.

3.3. RIP investigation

As we explained earlier, $\mathbf{\Psi}$ and $\tilde{\mathbf{\Psi}}$ are proved to be the sparsifying bases for the SRL and the SRLC. Having satisfied the sparsity property, the only issue that should be assessed to guarantee a high quality reconstruction is the mutual incoherence between the columns of $\mathbf{\Psi}$ and $\tilde{\mathbf{\Psi}}$ or alternatively the RIP. One way to approach the problem is following the same trend as explained in [13] because our channel coefficients can often be considered as drawn from a random distribution (such as Rayleigh). As is well-documented in the literature [20], for $K = 1, 2, \ldots$ the RIP constant δ_K of a matrix $\mathbf{A}$ (with normalized columns) is the smallest number that satisfies

$$-\delta_K \leq \frac{\|\mathbf{A}\mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2} - 1 \leq \delta_K, \tag{19}$$

for all K-sparse $\mathbf{x} \in \mathbb{R}^N$. Roughly speaking, as long as $0 < \delta_K < 1$, the RIP holds. In [13], by exploiting the effect of the random channel coefficients it is shown that if $M = O(K \log(N/K))$ the probability that there exists a K-sparse vector that satisfies $|\|\mathbf{A}\mathbf{x}\|_2^2 / \|\mathbf{x}\|_2^2 - 1| > \delta_K$ for a $0 < \delta_K < 1$ tends to 0, which means that with a high probability the RIP is satisfied. The same holds in our case for $\mathbf{\Psi}$. As an alternative, we have tried to numerically investigate the RIP property of the proposed fingerprinting maps to illustrate that the reconstruction will indeed have a high quality. To this aim, we can use the computationally less demanding definition in [21] where δ_K is defined as the maximum distance from 1 of all the eigenvalues of the $\binom{N}{K}$ submatrices, $\mathbf{A}_{\Lambda}^H \mathbf{A}_{\Lambda}$, derived from $\mathbf{A}$, where Λ is a set of indices with cardinality K which selects those columns of $\mathbf{A}$ indexed by Λ. It means that for each K, the RIP constant is given by

$$\delta_K = \max(|\lambda_{\max}(\mathbf{A}_{\Lambda}^H \mathbf{A}_{\Lambda}) - 1|,\ |\lambda_{\min}(\mathbf{A}_{\Lambda}^H \mathbf{A}_{\Lambda}) - 1|). \tag{20}$$

For the sake of computational feasibility, we consider the case where $M=5$, and hence $M^2=25$, and $N=36$ to generate a typical $\boldsymbol{\Psi}$ and $\tilde{\boldsymbol{\Psi}}$ using the other parameters adopted in Section 7. For such a case, we have computed the δ_K with $K=1,\ldots,6$ for $\boldsymbol{\Psi}$ and $\tilde{\boldsymbol{\Psi}}$. We also compute the δ_K for matrices with the same size containing elements drawn from a random normal distribution, i.e., $\mathbf{N}_{5\times 36}$ and $\mathbf{N}_{25\times 36}$. Note that such random matrices are proved to be a good choice in terms of the RIP and that is why we use them as a benchmark. In order to slightly heal the RIP, we apply the orthonormalization operation proposed in [9,16] to all the matrices before testing the RIP. The results are presented in Table 1. As is clear from the table, our proposed fingerprinting map for the SRL $\boldsymbol{\Psi}$ performs almost similar to $\mathbf{N}_{5\times 36}$ and loosely satisfies the RIP up to $K=2$. However, for $K>2$, δ_K starts increasing for both of them. Interestingly, we observe that for $\tilde{\boldsymbol{\Psi}}$ (also for $\mathbf{N}_{25\times 36}$) the RIP is met for K up to 6, which shows a considerable improvement as compared to $\boldsymbol{\Psi}$.

4. Exploiting additional time domain information (SRLC-TD)

For a fixed network, with known locations of the APs and GPs, the maximum delay difference can be computed during the initialization phase. It can be expressed by

$$\Delta\tau_{\max}=\max_{m,m',n}(|\tau_{m,n}-\tau_{m',n}|)=\max_{m,m',n}\left(\left|\frac{d(\mathrm{AP}_m,\mathrm{GP}_n)-d(\mathrm{AP}_{m'},\mathrm{GP}_n)}{\nu}\right|\right), \tag{21}$$

where $d(\cdot)$ denotes the Euclidean distance and ν is the velocity of signal propagation. As a result, the maximum delay difference experienced by any signal from a multipath channel is $\Delta\tau_{\max}+\gamma$, where γ denotes the maximum delay spread of the multipath channel. In principle, what we do in Section 3.2 is to compute the autocorrelations as well as the cross-correlations or in other words the zeroth lag of the autocorrelation and cross-correlation functions. Let us start by explaining what happens when we consider the complete autocorrelation function at each AP and the complete cross-correlation functions of the received signals at the different APs (SRLC-TD). As can be seen in Fig. 2, for a multipath channel, when we consider the complete autocorrelation function, the output is non-zero within the time span $[-1/B-\gamma,+1/B+\gamma]$. This means that there is potential information present in other lags than the zeroth lag which could further be exploited. Similarly, for the cross-correlations at the different APs, depending on the location of the SNs, we have to scan the time span $[-\Delta\tau_{\max}-1/B-\gamma,\Delta\tau_{\max}+1/B+\gamma]$ to make sure that we have at least some non-zero $r(\tau)$ values. Particularly, here we are interested in the elements of

$$\tilde{\mathbf{y}}^{(n)}=\mathbb{E}\{\mathbf{x}(t)\otimes\mathbf{x}(t-nT_s)^*\}, \tag{22}$$

which are given by

$$\begin{aligned} y^{(n)}_{m,m'} &= \mathbb{E}\{x_m(t)x^*_{m'}(t-nT_s)\}\\ &= \sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{l'=1}^{L} h_{l,m,k}h^*_{l',m',k}r(\tau_{l',m,k}-\tau_{l,m',k}+nT_s)\eta_k+\sigma_n^2\delta_n\delta_{m-m'}, \end{aligned}$$

Table 1
RIP test.

Matrix	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6
$\mathbf{N}_{5\times 36}$	0	0.8696	1.6167	2.2038	2.7692	3.2472
$\boldsymbol{\Psi}$	0	0.9820	1.7978	2.5670	3.2589	3.8070
$\mathbf{N}_{25\times 36}$	0	0.3442	0.5362	0.6581	0.7291	0.7793
$\tilde{\boldsymbol{\Psi}}$	0	0.5069	0.6966	0.8065	0.8609	0.9408

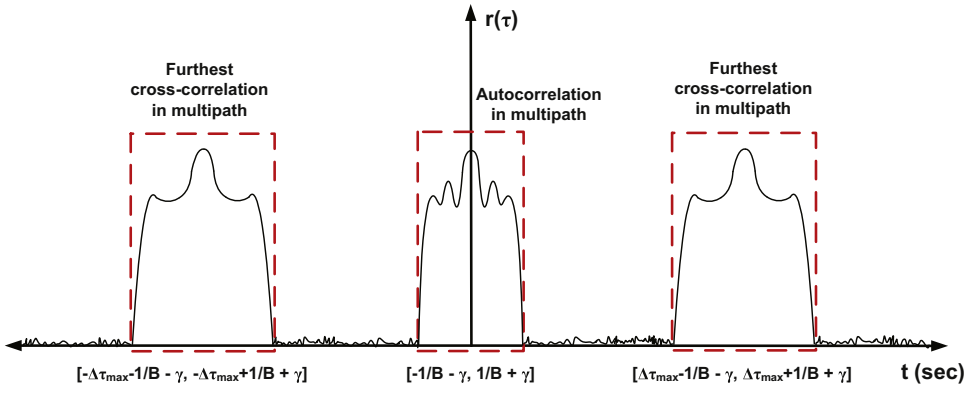

Fig. 2. Autocorrelations and cross-correlations in a multipath environment.

where T_s is the smallest time fraction in the system which in practice will be the sampling time since we implement the algorithms using temporal averaging. Therefore, we take $N_s = 1/(T_s B)$ samples per inverse bandwidth. Accordingly, by omitting the intermediate steps similar to the SRLC, we can compute the fingerprinting map for $\tilde{\mathbf{y}}^{(n)}$ as

$$(\tilde{\boldsymbol{\Psi}}^{(n)})^T = \sum_{l=1}^{L}\sum_{l'=1}^{L}\left\{\begin{bmatrix} h_{l,1,1}^{g*}h_{l',1,1}^{g}r(\tau_{l,1,1}^{g}-\tau_{l',1,1}^{g}+nT_s) & \cdots & h_{l,M,1}^{g*}h_{l',M,1}^{g}r(\tau_{l,M,1}^{g}-\tau_{l',M,1}^{g}+nT_s) \\ h_{l,1,2}^{g*}h_{l',1,2}^{g}r(\tau_{l,1,2}^{g}-\tau_{l',1,2}^{g}+nT_s) & \cdots & h_{l,M,2}^{g*}h_{l',M,2}^{g}r(\tau_{l,M,2}^{g}-\tau_{l',M,2}^{g}+nT_s) \\ \vdots & \ddots & \vdots \\ h_{l,1,N}^{g*}h_{l',1,N}^{g}r(\tau_{l,1,N}^{g}-\tau_{l',1,N}^{g}+nT_s) & \cdots & h_{l,M,N}^{g*}h_{l',M,N}^{g}r(\tau_{l,M,N}^{g}-\tau_{l',M,N}^{g}+nT_s) \end{bmatrix}\right\}. \tag{23}$$

As a result, we could consider all lags $n\in\{-N_s-\lceil\gamma/T_s\rceil,\ldots,N_s+\lceil\gamma/T_s\rceil\}$ of the complete auto-correlation functions and all lags $n\in\{\lfloor-(\Delta\tau_{\max}+\gamma)/T_s\rfloor-N_s,\ldots,\lceil(\gamma+\Delta\tau_{\max})/T_s\rceil+N_s\}$ of the complete cross-correlation functions. This way we will compute the autocorrelations for $N^{ac}=2(N_s+\lceil\gamma/T_s\rceil)$ lags whereas we have to compute cross-correlations for $N^{cc}=2(N_s+\lceil\Delta\tau_{\max}/T_s\rceil+\lceil\gamma/T_s\rceil)$ lags. Here, for the sake of simplicity of notation, we also assume that we compute N^{cc} autocorrelation lags and set the value of the autocorrelation function for the remaining $N^{cc}-N^{ac}=2\lceil(\Delta\tau_{\max})/T_s\rceil$ lags to zero.

The additional time lags contain new information which was not used in the SRLC. To exploit this potential information, we propose to incorporate all lags by solving

$$\tilde{\mathbf{y}}_{\mathrm{TD}}=\tilde{\boldsymbol{\Psi}}_{\mathrm{TD}}\boldsymbol{\theta}+\mathbf{1}_{N^{cc}}\otimes(\delta_n\tilde{\mathbf{p}}_n), \tag{24}$$

where

$$\tilde{\mathbf{y}}_{\mathrm{TD}}=[(\tilde{\mathbf{y}}^{(\lfloor-(\Delta\tau_{\max}+\gamma)/Ts\rfloor-N_s)})^T,\ldots,(\tilde{\mathbf{y}}^{(\lceil(\gamma+\Delta\tau_{\max})/T_s\rceil+N_s)})^T]^T,$$

and

$$\tilde{\boldsymbol{\Psi}}_{\mathrm{TD}}=[(\tilde{\boldsymbol{\Psi}}^{(\lfloor-(\Delta\tau_{\max}+\gamma)/Ts\rfloor-N_s)})^T,\ldots,(\tilde{\boldsymbol{\Psi}}^{(\lceil(\gamma+\Delta\tau_{\max})/T_s\rceil+N_s)})^T]^T,$$

are the augmented versions of the measurement vectors and fingerprinting maps computed at the different time lags. Hence, this time $\tilde{\boldsymbol{\Psi}}_{\mathrm{TD}}$ is a $N^{cc}M^2\times N$ matrix and thus (24) can be solved using LASSO or classical LS if it is underdetermined or overdetermined, respectively. It is noteworthy that, depending on the computational complexity constraints, at the expense of the identifiability gain we can also consider the lags to be spaced by the symbol time $1/B$ instead of T_s which would result in a smaller number of lags.

5. Blind SRLC using frequency domain information (SRLC-FD)

Remember that for both SRLC and SRLC-TD, $r(\tau)$ should be the same and known for all the sources (A.4 in Section 2) to make us capable of measuring/computing the fingerprinting map. This imposes some a priori knowledge on the problem which might be lacking in some practical situations, and thus we are also interested in an approach which is blind to the $r_k(\tau)$'s. Here, we tackle the issue which is specified by A.5 in Section 2, while we also try to take advantage of the large bandwidth of the received signal to gain some extra information, similar to Section 4, and enhance the SRLC, this time by approaching the problem from the frequency domain (SRLC-FD).

Let us start by explaining an appropriate filter bank design which plays an important role in the following analysis. Assume that we do not have any knowledge about the $r_k(\tau)$'s. Instead, at each AP we can efficiently estimate the bandwidth of the total received signal using appropriate spectrum estimation techniques [22]; we call it B and for the sake of simplicity of exposure it is assumed to be the same at different APs. Next, we use a set of filters $\{f^{(q)}(t)\}_{q=1}^{Q}$ to divide B into $Q=\lceil B(\Delta\tau_{\max}+\gamma)\rceil$ adjacent subbands $\mathcal{B}^{(q)}=[(q-1)B/Q, qB/Q)$ with bandwidth B/Q. A schematic view of an arbitrary signal, channel and the filter bank is shown in Fig. 3. Notably, since $B/Q=B/\lceil B(\Delta\tau_{\max}+\gamma)\rceil<1/\gamma$ with $1/\gamma$ representing the approximate coherence bandwidth of the channel, the output of the q-th filter at the m-th AP experiences a flat fading

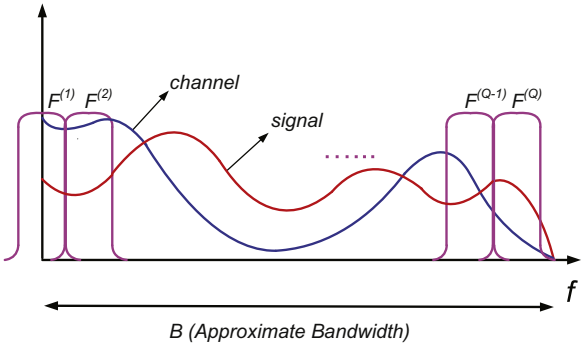

Fig. 3. Frequency domain filtering. $F^{(q)}$ denotes for the Fourier transform of $f^{(q)}(t)$.

channel $H_{m,k}^{(q)}$ for every SN_k. Therefore, the related output signal can be written as

$$x_m^{(q)}(t) = \sum_{k=1}^{K} [s_k(t) * f^{(q)}(t)] H_{m,k}^{(q)} + n_m(t) * f^{(q)}(t) = \sum_{k=1}^{K} s_k^{(q)}(t) H_{m,k}^{(q)} + n_m^{(q)}(t), \tag{25}$$

where $*$ denotes the convolution operator, and $s_k^{(q)}(t)$ and $n_m^{(q)}(t)$ respectively denote the filtered versions of $s_k(t)$ and $n_m(t)$. Further, by simply stacking the results for different APs, the total received signal vector can be expressed as $\mathbf{x}^{(q)}(t) = [x_1^{(q)}(t), \ldots, x_M^{(q)}(t)]^T$. Therefore, we have Q signals $\mathbf{x}^{(q)}(t)$ to compute $\tilde{\mathbf{y}}^{(q)} = \mathbb{E}\{\mathbf{x}^{(q)}(t) \otimes \mathbf{x}^{(q)*}(t)\}$ with its elements given by

$$y_{m,m'}^{(q)} = \mathbb{E}\{x_m^{(q)}(t) x_{m'}^{(q)*}\} = \mathbb{E}\left\{ \sum_{k=1}^{K} s_k^{(q)}(t) H_{m,k}^{(q)} \sum_{k'=1}^{K} s_{k'}^{(q)*}(t) H_{m',k'}^{(q)*} \right\} + \mathbb{E}\{n_m^{(q)}(t) n_{m'}^{(q)}(t)\} = \sum_{k=1}^{K} H_{m,k}^{(q)} H_{m',k}^{(q)*} \eta_k^{(q)} + \frac{\sigma_n^2}{Q} \delta_{m-m'}, \tag{26}$$

where the second equality follows from A.1 and A.3 and the last equality follows from A.2, as detailed in Section 2. Now, let us ignore the effect of the noise in (26) for the time being, and discover the fingerprints. Interestingly, owing to our proposed filtering, the location-dependent fingerprints $H_{m,k}^{(q)} H_{m',k}^{(q)*}$ do not depend on the $r_k(\tau)$'s and the effect of the different $r_k(\tau)$'s appears in the $\eta_k^{(q)}$'s, which can be handled within the sparse vector of interest. Now, if we consider that the sources can only be located on N GPs, we can use any training or analytical method to compute the $r_k(\tau)-$independent fingerprints at the m-th AP for the q-th subband as

$$\boldsymbol{\psi}_{m,m'}^{(q)} = [H_{m,1}^{(q)} H_{m',1}^{(q)*}, \ldots, H_{m,N}^{(q)} H_{m',N}^{(q)*}]^T. \tag{27}$$

As a result, (26) can be rewritten as

$$y_{m,m'}^{(q)} = (\boldsymbol{\psi}_{m,m'}^{(q)})^T \boldsymbol{\theta}^{(q)} + \frac{\sigma_n^2}{Q} \delta_{m-m'}, \tag{28}$$

where $\boldsymbol{\theta}^{(q)}$ is the K-sparse vector of interest for the q-th subband. The ensuing steps are similar to those of the SRLC and we can compute the fingerprinting map for $\tilde{\mathbf{y}}^{(q)}$ as

$$(\tilde{\boldsymbol{\Psi}}^{(q)})^T = \begin{bmatrix} |H_{1,1}^{(q)g}|^2 & H_{1,1}^{(q)g} H_{2,1}^{(q)g*} & \cdots & |H_{M,1}^{(q)g}|^2 \\ |H_{1,2}^{(q)g}|^2 & H_{1,2}^{(q)g} H_{2,2}^{(q)g*} & \cdots & |H_{M,2}^{(q)g}|^2 \\ \vdots & \vdots & \ddots & \vdots \\ |H_{1,N}^{(q)g}|^2 & H_{1,N}^{(q)g} H_{2,N}^{(q)g*} & \cdots & |H_{M,N}^{(q)g}|^2 \end{bmatrix}.$$

Now, based on this analysis, depending on the statistical properties of the received signals, i.e., spectrum of the $s_k(t)$'s, the following three cases can happen.

5.1. Flat spectrum

Looking at Fig. 3, we understand that if the spectrum of the sum of the $s_k(t)$'s is (almost) flat, the $\eta_k^{(q)}$'s will be (almost) the same in the different frequency bands $\mathcal{B}^{(q)}$. This basically makes it possible to construct an augmented version of the measurements as well as the fingerprinting maps, as $\eta_k^{(q)} \approx \eta_k$ will appear again in $\boldsymbol{\theta}^{(q)} = \boldsymbol{\theta}$ for all q. This means $\boldsymbol{\theta}$ will be a K-sparse signal with all elements equal to zero except for K elements equal to η_k. Thus, the ensuing steps are similar to those of the SRLC-TD as by constructing the augmented version of the run-time measurements as $\tilde{\mathbf{y}}_{\mathrm{FD}} = [(\tilde{\mathbf{y}}^{(1)})^T, \ldots, (\tilde{\mathbf{y}}^{(Q)})^T]^T$ and the one of the fingerprinting maps as $\tilde{\boldsymbol{\Psi}}_{\mathrm{FD}} = [(\tilde{\boldsymbol{\Psi}}^{(1)})^T, \ldots, (\tilde{\boldsymbol{\Psi}}^{(Q)})^T]^T$. Finally, we solve

$$\tilde{\mathbf{y}}_{\mathrm{FD}} = \tilde{\boldsymbol{\Psi}}_{\mathrm{FD}} \boldsymbol{\theta} + \mathbf{1}_Q \otimes \tilde{\mathbf{p}}_n. \tag{29}$$

As we explained, this time $\tilde{\boldsymbol{\Psi}}_{\mathrm{FD}}$ is a $QM^2 \times N$ matrix and thus (29) can be solved using LASSO or classical LS if it is underdetermined or overdetermined, respectively. It is worth pointing out that even for the case where the signals have a partially flat spectrum, we can design the filters for that flat part of the spectrum and again construct (29) where in such a case we will have less subbands.

5.2. Varying spectrum; the simple solution $Q=1$

In contrast to the case where the signals have a flat spectrum, for the non-flat case, we cannot construct augmented versions of the measurements and the maps for a unique $\boldsymbol{\theta}$ and solve a linear system similar to (29). Particularly, because of the different $\eta_k^{(q)}$'s in the different bands, the $\tilde{\boldsymbol{\Psi}}^{(q)}$'s and $\tilde{\mathbf{y}}^{(q)}$'s are related to different $\boldsymbol{\theta}^{(q)}$'s. In this case, as a straightforward

solution, we can simply take one of the bands, for instance the first band $\mathcal{B}^{(1)}$, and solve

$$\tilde{\mathbf{y}}^{(1)} = \tilde{\boldsymbol{\Psi}}^{(1)}\boldsymbol{\theta}^{(1)} + \tilde{\mathbf{p}}_n. \tag{30}$$

This way, we at least have the same identifiability gain as SRLC, but more importantly, we are blind to the $r_k(\tau)$'s. However, we still have some information present in the adjacent subbands which has not been exploited. This motivates the following subsection.

5.3. Varying spectrum; enhancing the identifiability gain

The question is how we can exploit the information present in all the subbands to attain an identifiability gain. An important observation which helps us to develop a solution is the fact that even though different subbands lead to different $\eta_k^{(q)}$'s for a non-flat spectrum, all the bands construct linear models, similar to (30), where in all of them the sparse $\boldsymbol{\theta}^{(q)}$'s share a common support, i.e., the support of $\boldsymbol{\theta}^{(q)}$ is the same $\forall q$. This important property motivates a group-LASSO (G-LASSO) type of solution to incorporate all the bands. However, note that different from classical G-LASSO, we have different maps $\tilde{\boldsymbol{\Psi}}^{(q)}$ for different subbands. Similar cases occur in the framework of the multiple measurement vectors (MMV) problem [23]. To deal with this, we propose a modified version of G-LASSO as defined by

$$\hat{\boldsymbol{\Theta}} = \arg\min_{\boldsymbol{\Theta}} \sum_{q=1}^{Q} \|\tilde{\mathbf{y}}^{(q)} - \tilde{\boldsymbol{\Psi}}^{(q)}[\boldsymbol{\Theta}]_{:,q}\|_2^2 + \lambda \sum_{n=1}^{N} \|[\boldsymbol{\Theta}]_{n,:}\|_2, \tag{31}$$

where $\boldsymbol{\Theta} = [\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(Q)}]$. The first term on the right hand side of (31) is the LS part which minimizes the error for the different subbands and the second term enforces group sparsity. It is worth pointing out that an analysis of the algorithms to solve (31) is outside the scope of this paper and here we restrict ourselves to standard convex optimization tools such as CVX [24] to solve the problem. Based on the discussions presented in [23] for MMV, incorporating all the subbands within (31) will result in a gain in terms of identifiability compared to (30), as is also corroborated by our simulation results in Section 7.

6. Improved localization using finite-alphabet sparsity

In particular cases where the SNs have a known equal signal power ($\eta_k = \eta, \forall k$) we can accommodate η within $\boldsymbol{\Psi}$ (or $\tilde{\boldsymbol{\Psi}}$, $\tilde{\boldsymbol{\Psi}}_{\text{TD}}$ and $\tilde{\boldsymbol{\Psi}}_{\text{FD}}$) and therefore $\boldsymbol{\theta}$ will be a K-sparse vector with 0 everywhere except for K elements which are 1. This means that our sparse vector (to be reconstructed using LASSO) has a finite-alphabet property which is not included in the optimization problem. Incorporating this extra information can help to improve the reconstruction quality and hence the localization performance for SRL (likewise, SRLC, SRLC-TD and SRLC-FD). The problem of sparse reconstruction under finite-alphabet constraints is investigated in [25,26]. In [26], efficient algorithms for multiuser detection (MUD) under sparsity and finite-alphabet constraints are developed. More general, sparse reconstruction under finite-alphabet constraints is investigated in [25] through two different approaches; sphere decoding and semi-definite relaxation (SDR), with a main emphasis on the former approach. Here, we re-derive and employ the SDR-based approach. Interestingly, when the alphabet set is $\{0,1\}$, $\|\boldsymbol{\theta}\|_0 = \|\boldsymbol{\theta}\|_1 = \|\boldsymbol{\theta}\|_2^2$ and $\|\boldsymbol{\theta}\|_1 = \boldsymbol{\theta}^T\mathbf{1} = \mathbf{1}^T\boldsymbol{\theta}$. This helps us to rewrite (9) (similarly also (17)) as

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\{0,1\}^N} \|\mathbf{y} - \boldsymbol{\Psi}\boldsymbol{\theta}\|_2^2 + \frac{\lambda-\epsilon}{2}\left(\boldsymbol{\theta}^T\mathbf{1} + \mathbf{1}^T\boldsymbol{\theta}\right) + \epsilon\|\boldsymbol{\theta}\|_2^2, \tag{32}$$

where $0 < \epsilon \le \lambda$. We can express the right-hand-side of (32) in a quadratic form as

$$J(\boldsymbol{\theta}) = \underbrace{\begin{bmatrix}\boldsymbol{\theta}\\ 1\end{bmatrix}^T \begin{bmatrix}\boldsymbol{\Psi}^H\boldsymbol{\Psi} + \epsilon\mathbf{I} & -\boldsymbol{\Psi}^H\mathbf{y} + \frac{\lambda-\epsilon}{2}\mathbf{1}\\ -\mathbf{y}^H\boldsymbol{\Psi} + \frac{\lambda-\epsilon}{2}\mathbf{1}^T & \mathbf{y}^H\mathbf{y}\end{bmatrix}}_{\mathbf{Q}_\theta} \underbrace{\begin{bmatrix}\boldsymbol{\theta}\\ 1\end{bmatrix}}_{\tilde{\boldsymbol{\theta}}}. \tag{33}$$

Note that minimizing (33) is a Boolean quadratic programming problem which permits several efficient algorithms including the quasi-maximum-likelihood SDR of [27]. However, to be able to employ SDR we have to express $J(\boldsymbol{\theta})$ as a function of $\boldsymbol{\alpha} = 2\boldsymbol{\theta} - \mathbf{1} \in \{-1, 1\}^N$. More specifically, after some simplifications we can write $J(\boldsymbol{\theta})$ as $I(\boldsymbol{\alpha}) = \tilde{\boldsymbol{\alpha}}^T\mathbf{Q}_\alpha\tilde{\boldsymbol{\alpha}}$ with $\tilde{\boldsymbol{\alpha}} = [\boldsymbol{\alpha}^T, 1]^T$ and

$$\mathbf{Q}_\alpha = \begin{bmatrix}(\boldsymbol{\Psi}^H\boldsymbol{\Psi} + \epsilon\mathbf{I})/4 & (\boldsymbol{\Psi}^H\boldsymbol{\Psi} + \lambda\mathbf{I})\mathbf{1}/4 - \boldsymbol{\Psi}^H\mathbf{y}/2\\ \mathbf{1}^T(\boldsymbol{\Psi}^H\boldsymbol{\Psi} + \lambda\mathbf{I})/4 - \mathbf{y}^H\boldsymbol{\Psi}/2 & \mathbf{1}^T(\boldsymbol{\Psi}^H\boldsymbol{\Psi} + \lambda\mathbf{I})\mathbf{1}/4 - \mathbf{y}^H\boldsymbol{\Psi}\mathbf{1}/2 - \mathbf{1}^T\boldsymbol{\Psi}^H\mathbf{y}/2 + \mathbf{y}^H\mathbf{y}\end{bmatrix}. \tag{34}$$

After relaxing the rank-1 constraint on $\tilde{\mathbf{A}}$ ($\tilde{\mathbf{A}} = \tilde{\boldsymbol{\alpha}}\tilde{\boldsymbol{\alpha}}^T$), we solve the following semi-definite programming (SDP) problem:

$$\begin{aligned}&\min_{\tilde{\mathbf{A}}} \quad \text{tr}(\mathbf{Q}_\alpha\tilde{\mathbf{A}})\\ &\text{s.t.} \quad \tilde{\mathbf{A}} \succcurlyeq \mathbf{0},\\ &\quad [\tilde{\mathbf{A}}]_{i,i} = 1, \quad i = 1, \ldots, N.\end{aligned}$$

The next step will be to factorize $\tilde{\mathbf{A}}$ to estimate the best $\tilde{\boldsymbol{\alpha}}$ via randomization as explained in [27]. Next, $\hat{\boldsymbol{\theta}}$ can simply be calculated using $\hat{\boldsymbol{\theta}} := (\tilde{\boldsymbol{\alpha}} + \mathbf{1})/2$. We expect that including this unused information (finite-alphabet sparsity) within our reconstruction model leads to a performance gain, as is validated by our simulation results. This basically motivates using this model for reconstructing a finite-alphabet sparse $\boldsymbol{\theta}$.

7. Numerical results

In this section, we investigate the performance of the proposed algorithms in terms of probability of detection (P_d), probability of false alarm (P_{fa}) and positioning root mean squared error (PRMSE) against $1/\sigma_n^2$, the number of existing SNs K and the number of GPs N.

To this aim, we consider a room of size $10 \times 10 \times 3$ m^3 even though our goal is to find the location of the sources on the floor (in 2-D) of size 10×10 m^2. This 2-D area is divided into $N = 100$ cells represented by their central GPs. The APs are randomly placed on the ceiling at a height of 3 m and our (up to $K = 10$) non-cooperative sources are considered to be on the floor at a height of 1.8 m. Two different scenarios are considered where in the first scenario the sources are randomly placed but they are always on-grid whereas in the second scenario they can be located anywhere, i.e., they can also be off-grid.

The following assumptions about the signal, channel and measurements are respectively in place:

- We consider wideband BPSK signals with a rectangular pulse shape, 3 dB bandwidth of $B = 10$ MHz and power $\eta = 1$. This means $r(\tau) = 1 - |\tau|/B$ for the baseband equivalent signal. The carrier frequency for the passband signal is 2.4 GHz. For all simulations, $r(\tau)$ is assumed to be the same and fixed for all the sources, unless otherwise mentioned. We compute the autocorrelation and cross-correlation functions during a time-slot of length $T = 0.1$ ms. This is equal to recording $T \times B = 10^{-4} \times 10^7 = 1000$ BPSK symbols for our computations. Hence, even for moving sources with low dynamics, which is a realistic assumption for the networks under consideration, the length of the time-slot ($T = 0.1$ ms) will not put a large constraint on the dynamics of the sources.
- In order to assess the algorithms for a realistic channel model (with no simplifying assumptions), we use synthetic data from the RIR generator provided by [18] for the wireless system explained earlier.
- Instead of taking ideal expectations $\mathbb{E}\{\cdot\}$ in the measurement phase, we work with discrete-time signals of limited length and hence the computations of the autocorrelations as well as the cross-correlations will not be ideal as in the derivations of Section 3. As a result, the noise terms $n_m(t)$ will not be completely eliminated in the cross-correlations and they will be an approximation of what is considered for the autocorrelations, and therefore, this will slightly affect our performance. Likewise, the value of the autocorrelations and cross-correlations (in $\mathbf{y}$ or $\tilde{\mathbf{y}}$) will also be approximations of the ideal computations due to this finite-length error.

All simulations are averaged over $P = 100$ independent Monte Carlo (MC) runs where in each run the sources are deployed on different random locations. For all the reconstruction problems, we choose λ by cross-validation as explained in [28]. For the case of on-grid sources, we concentrate on the *detection* performance, i.e., we are only interested to know which elements of the estimated $\boldsymbol{\theta}$ correspond to a source and which elements are zeros, i.e., we only care about the support of $\boldsymbol{\theta}$. Based on this, we define P_{err}, P_d and P_{fa} as follows [29]:

- P_{err}:=the probability that a source is detected when the source is in fact *not* present or it is *not* detected when it is in fact present.
- P_d:=the probability that a source is detected when the source is in fact present.
- P_{fa}:=the probability that a source is detected when the source in fact *not* present.

Basically, P_d and P_{fa} specify all the probabilities of interest. However, we need a detection threshold to be able to compute them. To find the best threshold, we carry out a linear search within the range $[0, \max(\hat{\boldsymbol{\theta}}))$ and select the value which minimizes P_{err}. On the other hand, for off-grid sources we plot both P_d and the positioning root mean squared error (PRMSE) defined by

$$\text{PRMSE} = \sqrt{\frac{1}{PK} \sum_{p=1}^{P} \sum_{k=1}^{K} e_{k,p}^2},$$

where $e_{k,p}$ represents the distance between the real location of the k-th source and its estimated location at the p-th MC trial.

Finally, we would like to point out that we do not compare our results with the KNN, the BC, or even semi-definite relaxation (SDR)-based algorithms because the superiority of the ℓ_1-norm minimization approaches (at least for the SRL) compared to KNN, BC and SDR-based algorithms is respectively illustrated in [8,9,15]. Instead, the SRL will be used as the benchmark multi-source RSS-based localization algorithm.

7.1. Performance evaluation with $M=15$ APs

We start by investigating the performance of the proposed algorithms for the case that there are $M=15$ APs. In this case, $M^2=225>N$, and therefore, the SRLC is expected to perform very well and be capable of recovering $\boldsymbol{\theta}$ with LS too. Note that here LS refers to the classical LS applied within the framework of the SRLC. For the sake of simplicity, we consider the SNs to have equal power, i.e., $\eta_k=\eta\,\forall k$. This allows us to employ and assess the idea of finite-alphabet sparsity to recover $\boldsymbol{\theta}$, as well.

In the first simulation, we consider $K=10$ sources randomly located on the GPs. As is clear from the schematic view of Fig. 4 for $1/\sigma_n^2=20$ dB, while the SRL can only localize 3 sources, the proposed SRLC (solved by LASSO) and the SRLC (solved by LS) are capable of localizing all the sources. This has motivated us to assess the performance of the SRL solved by the finite-alphabet sparsity idea (we call it SRL-FA) and as is clear from the figure, SRL-FA could localize 4 sources which is improved compared to SRL. Obviously, this improvement also holds for the case of the SRLC with finite-alphabet sparsity; however, since the SRLC is already performing good enough, we do not plot those results. Note that in all the simulations with finite-alphabet sparsity $\epsilon=0.5\lambda$ and we perform 100 randomization trials.

In order to further investigate the performance of the aforementioned algorithms, we plot the detection and false alarm performance of the algorithms against $1/\sigma_n^2$ as well as the number of existing sources K. In Fig. 5, we assume $K=4$ sources. As is clear from the figure, the SRLC approaches (solved by LASSO and LS) perform very good as they attain $P_d=1$ and $P_{fa}\approx 0$

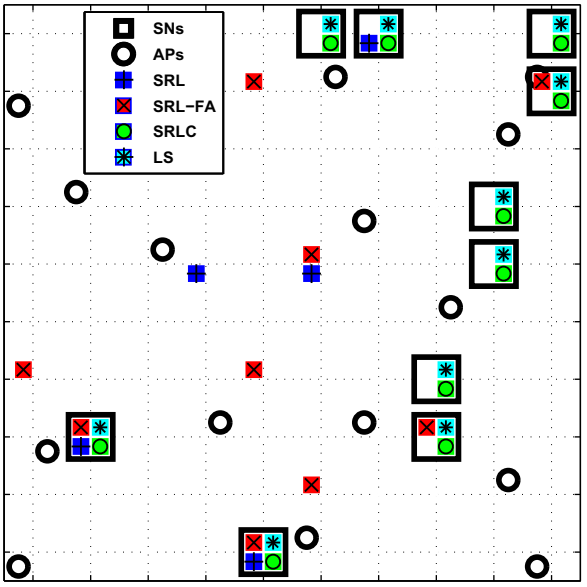

Fig. 4. Schematic view; $M=15$, $K=10$ and $1/\sigma_n^2=20$ dB.

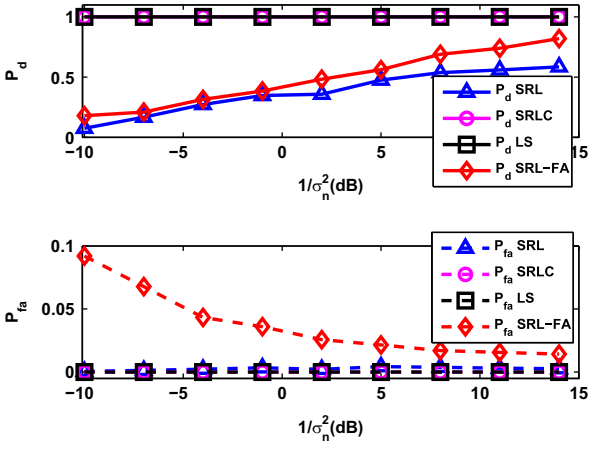

Fig. 5. Performance vs. $1/\sigma_n^2$ for $M=15$ and $K=4$.

for a large span of $1/\sigma_n^2$. The SRL-FA is clearly achieving a better P_d compared to SRL; however, it has a higher P_{fa} as well when its P_d is low. Notably, for all the algorithms, the general trend is an improvement with $1/\sigma_n^2$.

Now, let us get a better understanding by taking a look at the performance of the algorithms for $1/\sigma_n^2 = 20$ dB vs. K in Fig. 6. As can be seen, SRLC (in either case) can efficiently localize all the sources while for the SRL the performance drops by increasing the number of sources. The important observation here is that SRL-FA is almost capable of localizing up to $K=3$ sources with a very high P_d and minimum P_{fa} while this number reduces to $K=1$ for the stand-alone SRL. However, for $K > 3$ even though the P_d is always better for the SRL-FA, the P_{fa} also increases. Based on the observations in Figs. 5 and 6, we can conclude that the finite-alphabet sparsity idea is useful for the range $K \leq 3$ in this setup. At this point, it is noteworthy that we do not plot the results for $K > 10$ sources since for those cases $\boldsymbol{\theta}$ is not really sparse, i.e., we do not have $K \ll N$.

7.2. Further improvement with $M=5$ APs and blindness to $r(\tau)$

In this subsection, we consider the case where we have only $M=5$ APs available. For such a case, $M^2 = 25 < N$ and thus it is expected that even the SRLC might not be capable of localizing all the $K=10$ sources. This basically motivates employing the SRLC-TD to incorporate other time lags and hopefully improve the performance over the proposed SRLC. Moreover, this subsection is also meant to investigate the performance of the SRLC-FD algorithm. To this aim, we assume that all the sources have different η_k's with a uniform distribution in the range of $[0.8, 1.2]$ and we assume that $r(\tau)$ is unknown to

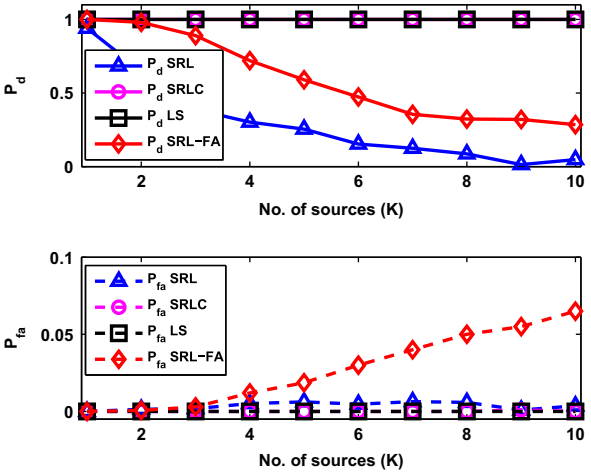

Fig. 6. Performance vs. K for $M=15$ and $1/\sigma_n^2 = 20$ dB.

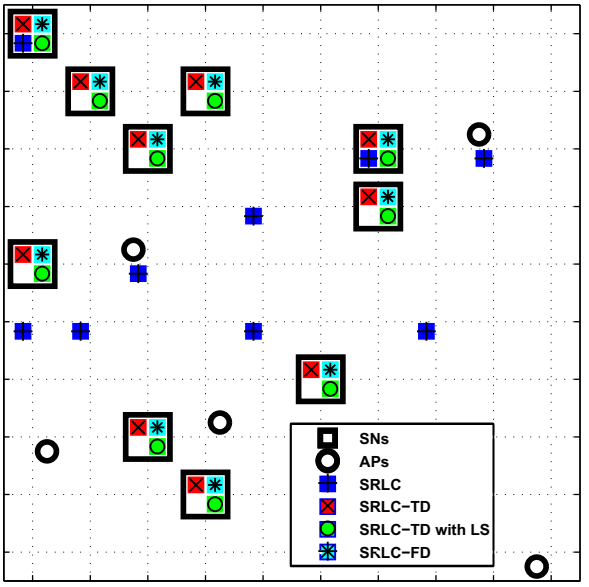

Fig. 7. Schematic view; $M=5$, $K=10$ and $1/\sigma_n^2 = 20$ dB.

SRLC-FD. This calls for a different fingerprinting map as explained in Section 5. We would like to emphasize that SRLC-FD can be employed even for cases where all the sources have different $r_k(\tau)$'s. However, since this cannot be handled by the SRLC and the SRLC-TD, we omit those results here.

Similar to the previous subsection, we consider $K=10$ sources randomly located on the GPs. Fig. 7 depicts a schematic view of localization for $1/\sigma_n^2 = 20$ dB. As can be seen, while SRLC is only capable of localizing $K=2$ sources, the other three enhanced algorithms, i.e., SRLC-TD (solved with LASSO), SRLC-TD (solved with LS) and the blind algorithm (SRLC-FD) could localize all the sources simultaneously. Notably, for the sake of a lower computational complexity, we consider only 6 time lags for the SRLC-TD which are spaced by $1/(2B)$ ($>T_s$) in our simulations. For the SRLC-FD, we have designed $Q=10$ filters and the proposed G-LASSO solution (explained in Section 5.3) is employed. It is also worth mentioning that since all the sources have different η_k's, finite-alphabet sparsity is not applicable in this subsection.

As in the previous subsection, we would also like to further assess the proposed algorithms in terms of P_d and P_{fa}. Fig. 8 compares the performance of the aforementioned algorithms against $1/\sigma_n^2$ for $K=4$. SRLC-FD ($Q=1$) denotes the idea of exploiting only one frequency band as explained in Section 5.2. As is clear from the figure, SRLC-FD ($Q=1$) is performing very close to SRLC while it is blind to $r(\tau)$. Interestingly, SRLC-FD is performing better than SRLC while it is blind. Notably, SRLC-TD (solved with LASSO) and SRLC-TD (solved with LS) both are performing good and attain the best possible performance for $1/\sigma_n^2$ values larger than -1 dB. This observation that SRLC-TD is less affected by noise can be justified by referring to (24) where we have shown that only measurements in the zeroth time lag are contaminated with noise and the rest of the lags are almost clean.

Let us get a more complete picture of the performance of the algorithms by taking a look at Fig. 9 where the detection and false alarm probabilities are depicted against K for $1/\sigma_n^2 = 20$ dB. As can be seen, the performance drops for the SRLC and the SRLC-FD ($Q=1$) with K and thus P_d starts decreasing whereas P_{fa} rises for $K>3$. Interestingly, for a large enough

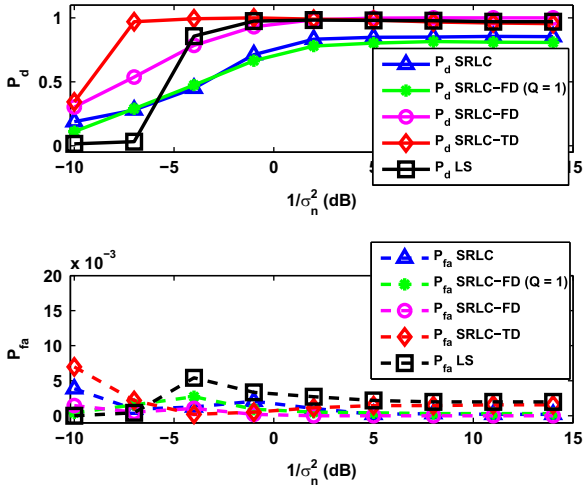

Fig. 8. Performance vs. $1/\sigma_n^2$ for $M=5$ and $K=4$.

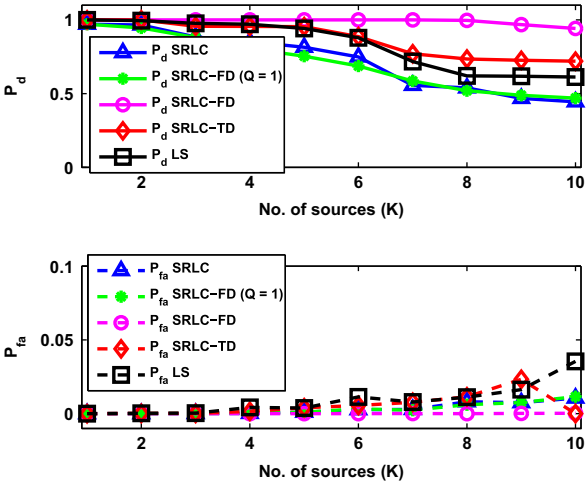

Fig. 9. Performance vs. K for $M=5$ and $1/\sigma_n^2 = 20$ dB.

$1/\sigma_n^2$ (i.e., small enough noise), SRLC-FD attains an optimal performance even for K up to 10. This result corroborates the fact that our blind algorithm with no information about $r(\tau)$, by exploiting the information of the $Q=10$ frequency subbands could outperform SRLC in terms of the number of identifiable sources. Note that there is a major improvement in SRLC-FD compared to SRLC-FD ($Q=1$). The SRLC-TD (both with LASSO and LS) starts degrading for $K \geq 5$ which can indeed be improved at the expense of complexity by increasing the number of time lags if the signal and channel properties permit.

7.3. Performance evaluation for off-grid sources

In this subsection, we intend to investigate the effect of off-grid sources on the performance of the proposed localization paradigm. Having assessed the improvements by exploiting time lags and frequency domain information via respectively SRLC-TD and SRLC-FD, here we only concentrate on the primary algorithm SRLC. Notably, the following off-grid experiments also demonstrate the performance of the SRLC when the measurements are inconsistent with the fingerprinting map. In an off-grid scenario, we expect to observe non-zero values in $\hat{\boldsymbol{\theta}}_{\text{SRLC}}$ corresponding to the GPs around an off-grid source if the channels observed by the neighboring grid points are correlated with the measurements. In order to increase this regional correlation, we should work at lower frequencies and that is why for the following simulations $f_c=100$ MHz and $B=1$ MHz. This means that for the same number of BPSK symbols as before, we have to record $T=1000/10^6=1$ ms of the received signals. This is shown in Fig. 10 where we depict a 3-D snapshot of $\hat{\boldsymbol{\theta}}_{\text{SRLC}}$ for $M=7$, $N=196$, $K=3$ and $1/\sigma_n^2=20$ dB. As can be seen, mostly the GPs around the sources return non-zero values which helps us to localize the off-grid sources. Now that we can have continuous locations of the sources in the 2-D area of interest, it makes sense to also plot the PRMSE of our estimates where we only constrain ourselves to finding the nearest GP to the off-grid sources. To further elaborate on the performance, we also plot P_d where a source is considered to be detected if it is estimated to be in a circle with a radius of $\sqrt{2}$ around its real location. To this aim, for the sake of picturing out irrelevant location estimates to achieve a meaningful PRMSE estimate, we consider that we know K and that is why we omit P_{fa} curves. The rest of the parameters is the same as in previous simulations, unless otherwise mentioned.

Fig. 11 illustrates the performance against $1/\sigma_n^2$ for $M=7$ APs with $K=1$ and 3 SNs randomly located on the floor (at a height of 1.8 m) of the room. As can be seen, for a single-source scenario the PRMSE goes below 1 m (the cell size) and this means the source can be very-well localized as is corroborated by the corresponding P_d curve. However, for $K=3$ SNs PRMSE and P_d are slightly degraded. It is worthy of being noted that for the multiple off-grid source localization, the more distant the sources are, the better we can relate the nonzero values of the estimated $\boldsymbol{\theta}$ to the closest GP. This shows a shortcoming of SRLC for localizing off-grid sources which constrains us to artificially avoid the sources to be located in neighboring cells.

Further, in Fig. 12, we try to investigate the performance of the SRLC against $N=36$, 64, 100, 144, 196, 324, 484, 676 and 900 while the room size is kept fixed. The main intention is to assess how an increased correlation between the GPs affects the performance. Note that, however, for a fair comparison in terms of reconstruction (and hence localization), we should also keep the ratio M/N (sometimes called compression rate) constant. In this simulation, we keep a fairly reasonable ratio $M/N=1/4$. As can be seen, the results are plotted for two different noise levels $1/\sigma_n^2=-5$ dB and 5 dB. As expected the performance is relatively better in the lower noise level. However, even with $N=900$, the correlation between the columns of the dictionary is not so severe to spoil the reconstruction, and the performance keeps improving with N. We would like to highlight though that further increasing K will indeed lead to a situation where the RIP will be drastically affected and SRLC

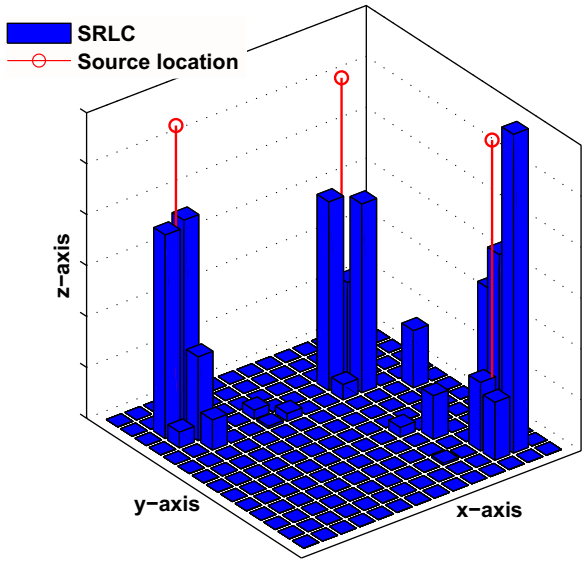

Fig. 10. 3-D view of $\hat{\boldsymbol{\theta}}_{\text{SRLC}}$ for $M=7$, $N=196$, $K=3$ and $1/\sigma_n^2=20$ dB.

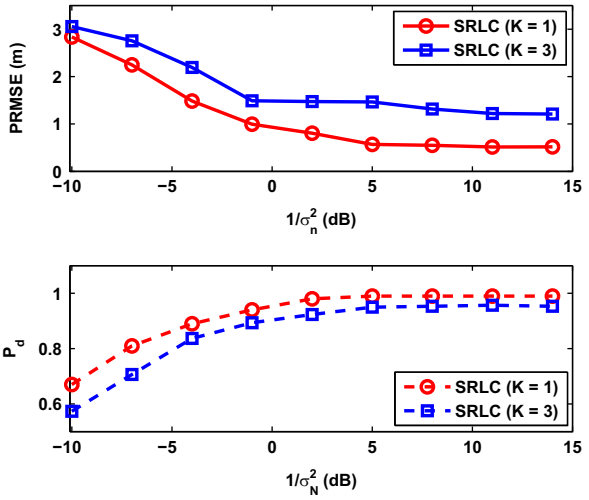

Fig. 11. Off-grid performance vs. $1/\sigma_n^2$; $M=7$, $K=1$ and 3.

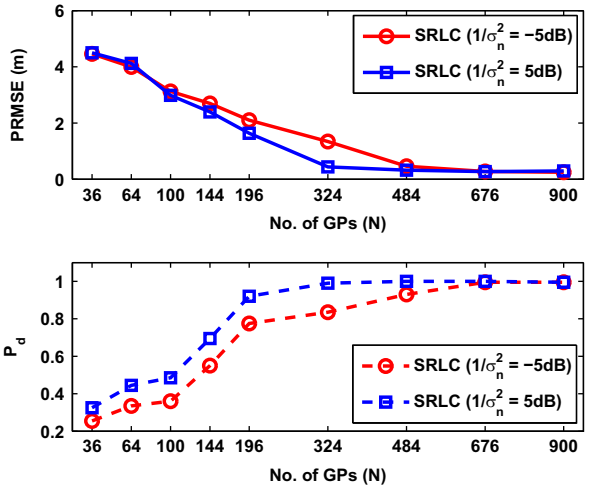

Fig. 12. Off-grid performance vs. N with $K=2$ for $1/\sigma_n^2 = -5$ dB and 5 dB.

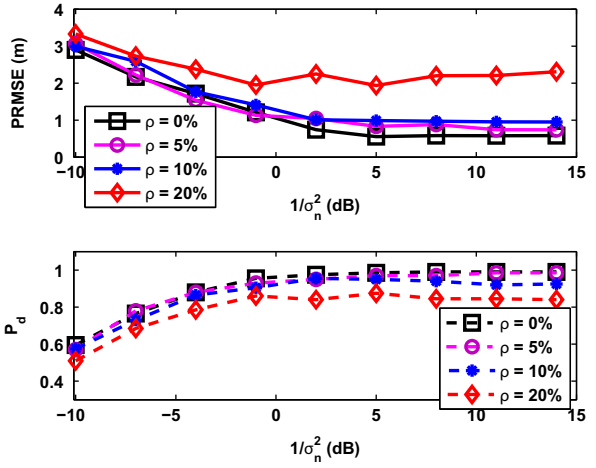

Fig. 13. Off-grid performance vs. $1/\sigma_n^2$ for $\rho = 0\%$, 5%, 15% and 20%.

will fail. In principle, this is an inherent limitation of any sparsity-aware localization algorithm which should be taken into account at the preliminary system level design.

Finally, we assess the sensitivity of the SRLC w.r.t. perturbations in the trained/computed fingerprinting map $\tilde{\mathbf{\Psi}}$. Such perturbations can for instance happen due to variations in the environment during the run-time phase. To this aim, a perturbation matrix $\mathbf{\Delta}$ drawn from a complex random Gaussian distribution is added to $\tilde{\mathbf{\Psi}}$. Accordingly, a perturbation ratio ρ is defined by $\rho = \|\mathbf{\Delta}\|/\|\tilde{\mathbf{\Psi}}\|$ which is set to 0% (no perturbation), 5%, 10% and 20% in our simulations. As can be seen from Fig. 13, the perturbations show their effect mostly in the lower $1/\sigma_n^2$'s. Particularly, for $K=2$, ρ's up to 10%, and $1/\sigma_n^2 \geq 5$ dB, the same localization accuracy (less than 1 m and $P_d=1$) as when there is no perturbation can be attained. However, increasing ρ to $\rho > 10\%$ leads to a performance degradation even for high $1/\sigma_n^2$'s. It is noteworthy that this experiment illustrates that our proposed idea can even work when all three model non-idealities simultaneously exist, i.e, measurement noise, off-grid sources and a slightly varying environment.

8. Computational complexity and conclusions

Before concluding this paper, we would like to comment on the complexity of the proposed approaches (SRLC, SRLC-TD, SRLC-FD) compared to the classical approach (SRL). Obviously, the enhanced source detection capability of the proposed approaches comes at a price and that is increased complexity. The proposed approaches (SRLC, SRLC-TD and SRLC-FD) respectively require a larger dictionary of size $M^2 \times N$, $N^{cc}M^2 \times N$, and $QM^2 \times N$ compared to the smaller one of SRL of size $M \times N$. Solving our sparse reconstruction problems using LASSO or similarly basis pursuit denoising (BPDN) using the approach of [30] for example requires a complexity that is linear in the number of rows of the dictionary. Therefore, the aforementioned algorithms are respectively M, MN^{cc}, and QM times more demanding in terms of computational cost than the SRL.

This paper studies the problem of localizing *multiple sources* using their RSS measurements in multipath environments. We have proposed a novel fingerprinting paradigm to exploit the information present in the cross-correlations of the received signals at the different APs which is ignored in existing sparsity-aware fingerprinting approaches. Besides, we have also proposed to further enhance the novel paradigm by incorporating other lags than the zeroth lag of the auto-correlation/cross-correlation functions. Moreover, we have extended our proposed idea to be able to operate when we are blind to the statistics of the source signals. Finally, we have employed the concept of finite-alphabet sparsity in our framework to deal with the sparse vectors of interest, if they contain finite-alphabet elements. Our extensive simulation results corroborate the efficiency of the proposed algorithms in terms of localization accuracy as well as detection capability.

References

[1] G. Mao, B. Fidan, B.D.O. Anderson, Wireless sensor network localization techniques, Comput. Netw. 51 (10) (2009) 2529–2553.
[2] B. Li, J. Salter, A.G. Dempster, C. Rizos, Indoor positioning techniques based on wireless LAN, in: Proceedings of the IEEE International Conference on Wireless Broadband and Ultra Wideband Communications, March 2006.
[3] D. Madigan, E. Elnahrawy, R.R. Matrin, P.K.W. Ju, A. Krishnakumar, Bayesian indoor positioning systems, in: Proceedings of the IEEE INFOCOM, March 2005, pp. 1217–1227.
[4] D.L. Donoho, Compressed sensing, IEEE Trans. Inform. Theory 52 (April) (2006) 1289–1306.
[5] V. Cevher, R.G. Baraniuk, Compressive sensing for sensor calibration, in: Proceedings of the IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM), July 2008.
[6] V. Cevher, M.F. Durate, R.G. Baraniuk, Distributed target localization via spatial sparsity, in: Proceedings of the European Signal Processing Conference (EUSIPCO), August 2009.
[7] C. Feng, S. Valaee, Z. Tan, Multiple target localization using compressive sensing, in: Proceedings of the IEEE GLOBECOM, November–December 2009, pp. 1–6.
[8] C. Feng, W.S.A. Au, S. Valaee, Z. Tan, Compressive sensing based positioning using RSS of WLAN access points, in: Proceedings of the IEEE INFOCOM, March 2010, pp. 1–9.
[9] C. Feng, W.S.A. Au, S. Valaee, Z. Tan, Received signal strength based indoor positioning using compressive sensing, IEEE Trans. Mob. Comput. 11 (December) (2012) 1983–1993.
[10] S. Nikitaki, P. Tsakalides, Localization in wireless networks via spatial sparsity, in: Proceedings of the Asilomar Conference on Signals, Systems and Computers, November 2010, pp. 236–239.
[11] S. Nikitaki, P. Tsakalides, Localization in wireless networks based on jointly compressed sensing, in: Proceedings of the European Signal Processing Conference (EUSIPCO), August–September 2011, pp. 1809–1813.
[12] G.T.D. Milioris, P. Jacquet, P. Tsakalides, Indoor positioning in wireless LANs using compressive sensing signal-strength fingerprinting, in: Proceedings of the European Signal Processing Conference (EUSIPCO), August–September 2011, pp. 1776–1780.
[13] B. Zhang, X. Cheng, N. Zhang, Y. Cui, Y. Li, Q. Liang, Sparse target counting and localization in sensor networks based on compressive sensing, in: Proceedings of IEEE INFOCOM, April 2011, pp. 2255–2263.
[14] C.R. Comsa, A.M. Haimovich, S.C. Schwartz, Y.H. Dobyns, J.A. Dabin, Source localization using time difference of arrival within a sparse representation framework, in: Proceedings of the IEEE ICASSP, May 2011, pp. 2872–2875.
[15] C.R. Comsa, A.M. Haimovich, S.C. Schwartz, Y.H. Dobyns, J.A. Dabin, Time difference of arrival based source localization within a sparse representation framework, in: Proceedings of the Conference on Information Sciences and Systems (CISS), March 2011, pp. 1–6.
[16] H. Jamali-Rad, G. Leus, Sparsity-aware multi-source TDOA localization, IEEE Trans. Signal Process. 61 (October) (2013) 4874–4887.
[17] H. Jamali-Rad, H. Ramezani, G. Leus, Sparse multi-target localization using cooperative access points, in: Proceedings of the IEEE Sensor Array Multichannel Processing Workshop (SAM), June 2012, pp. 353–356.
[18] E.A.P. Habets, Room Impulse Response Generator, 2008, Available at ⟨http://home.tiscali.nl/ehabets/rir_generator.html⟩.
[19] D. Malioutov, M. Çetin, A.S. Wilsky, A sparse signal reconstruction perspective for source localization with sensor arrays, IEEE Trans. Signal Process. 53 (August) (2005) 3010–3022.
[20] E.J. Candes, M.B. Wakin, An introduction to compressive sampling, IEEE Signal Process. Mag. 25 (March) (2008) 21–30.
[21] J.D. Blanchard, C. Cartis, J. Tanner, Compressed sensing: how sharp is the restricted isometry property? SIAM Rev. 53 (1) (2011) 105–125.

[22] C. Bingham, M.D. Godfrey, J.W. Tukey, Modern techniques of power spectrum estimation, IEEE Trans. Audio Electroacoust. 15 (June) (1967) 56–66.
[23] J. Chen, X. Huo, Theoretical results on sparse representations of multiple-measurement vectors, IEEE Trans. Signal Process. 54 (December) (2006) 4634–4643.
[24] CVX Research Inc., CVX: Matlab Software for Disciplined Convex Programming, Version 2.0 Beta, September 2012, ⟨http://cvxr.com/cvx⟩.
[25] Z. Tian, G. Leus, V. Lottici, Detection of sparse signals under finite-alphabet constraints, in: Proceedings of the IEEE ICASSP, May 2009, pp. 2349–2352.
[26] H. Zhu, G. Giannakis, Exploiting sparse user activity in multiuser detection, IEEE Trans. Commun. 59 (February) (2011) 454–456.
[27] T.N.D.W.-K. Ma, K.M. Wong, Z.-Q. Luo, P.-C. Ching, Quasi-ML multiuser detection using semi-definite relaxation with application to synchronous CDMA, IEEE Trans. Sig. Process. 50 (April) (2002) 912–922.
[28] R.R. Picard, R.D. Cook, Cross-validation of regression models, J. Am. Stat. Assoc. 79 (September) (1984) 575–583.
[29] S.M. Kay, Fundamentals of Statistical Signal Processing: Detection Theory, Prentice-Hall, Englewood Cliffs, NJ, 1993.
[30] P.R. Gill, A. Wang, A. Molnar, The in-crowd algorithm for fast basis pursuit denoising, IEEE Trans. Signal Process. 59 (October) (2011) 4595–4605.