Dynamic Learning for Cognitive Radio Sensing

Seung-Jun Kim
Dept. of Electrical & Computer Engr.
University of Minnesota
Minneapolis, MN 55455, U.S.A.
Email: seungjun@umn.edu

Georgios B. Giannakis
Dept. of Electrical & Computer Engr.
University of Minnesota
Minneapolis, MN 55455, U.S.A.
Email: georgios@umn.edu

***Abstract*—Spectrum sensing algorithms for cognitive radios that can interpolate and predict the spatio-temporal interference power distribution are proposed using the dictionary learning framework. The algorithms jointly estimate the dictionaries to capture the spatial spectrum measurements as well as their temporal dynamics via parsimoniously chosen atoms. Both batch and efficient online implementations are developed. Numerical tests verify the effectiveness of the novel approach.**

I. Introduction

The cognitive radio (CR) concept aims at increasing spectral efficiency of wireless communication systems via agile sensing of the environment and intelligent adaptation. Through spectrum overlay, in which the access priorities of primary user (PU) systems are respected by CR systems, the inefficiency of fixed spectrum allocation can be mitigated [1].

An essential component of CR systems is spectrum sensing, which refers to various tasks related to dynamically acquiring the status of spectral resources. Extensive research has been done on this topic, ranging from identifying the presence or the absence of PU transmitters [2], to revealing their number and locations, as well as estimating the relevant channel gains useful for subsequent resource allocation [3].

The goal of this work is to leverage cooperation among CRs dispersed over a geographical area to accurately assess the interference power levels at the node locations, as well as predict their future levels. An important challenge is that the CR network does not have prior information on the number of PU transmitters, and the corresponding PU-to-CR channel gains, which are essential for combining the measurements from different nodes. Furthermore, it is assumed that the CRs cannot always report their measurements to the fusion center, due to various practical reasons. Still, it is desired that the missing observations are reconstructed from the available ones, and their future values predicted.

A logistic regression model was employed to predict future spectrum occupancy in [4]. Time series models were used to aid channel switching decisions in [5]. Channel sensing and access schedules were determined in the framework of partially observed Markov decision process in [6].

In this work, contemporary tools from machine learning and compressive sensing are employed. Specifically, a dictionary learning framework is adopted to model the spatial RF interference distribution. Different from a related approach in [7], where the temporal dimension was incorporated simply by stacking observations at multiple time instants in a super-vector, the novel approach here advocates a vector autoregressive (AR) model for the sparse coefficients, where the unknown dynamic model is jointly estimated.

The rest of this paper is organized as follows. The system model and the problem statement are presented in Sec. II. A dictionary learning formulation for dynamic signals and a batch solution are provided in Sec. III. An online counterpart is proposed in Sec. IV. The results from numerical tests are presented in Sec. V, followed by conclusions in Sec. VI.

II. System Model and Problem Statement

A network of M CR nodes is deployed in a geographical area and over a frequency band where K PU transmitters operate. Suppose that the k-th PU transmits at time t with power level $p_k(t)$. Let $g_{mk}(t)$ represent the channel gain from the k-th PU transmitter to the m-th CR node. Then, the interference power $\pi_m(t)$ due to the K PUs experienced by CR m at time t can be expressed as

$$\pi_m(t) = \sum_{k=1}^{K} g_{mk}(t)p_k(t), \quad m = 1, 2, \ldots, M. \tag{1}$$

Due to various practical constraints such as task priorities, or the sleep modes for saving battery, not all CR nodes can take measurements in each time instant. Also, communication errors and congestions in the control channel may hinder timely relaying of the measurements to the fusion center. These considerations motivate the goal of spectrum sensing, which is to interpolate the interference levels at all CR node locations, based on the *subset* of measurements collected at the fusion center at each time. Moreover, through learning the dynamics of the spatio-temporal variation of the interference levels, prediction of the future spectrum states is desired as well.

Define vectors $\boldsymbol{\pi}(t) := [\pi_1(t), \ldots, \pi_M(t)]^{\mathcal{T}}$ and $\mathbf{p}(t) := [p_1(t), \ldots, p_K(t)]^{\mathcal{T}}$, where $\cdot^{\mathcal{T}}$ denotes transposition. Also, define matrix $\mathbf{G}(t)$ with its (m, k)-entry equal to $g_{mk}(t)$. Then, (1) can be compactly written as

$$\boldsymbol{\pi}(t) = \mathbf{G}(t)\mathbf{p}(t). \tag{2}$$

This work was supported by NSF grants ECCS 1002180, ECCS 1202135 and AST 1247885.

978-1-4673-3146-3/13/$31.00 ©2013IEEE

Let $\mathcal{M}^{obs}(t) \subset \mathcal{M} := \{1, 2, \ldots, M\}$ denote the set of CR nodes whose observations have been collected. Stacking these observations in vector $\mathbf{y}^{obs}(t) \in \mathbb{R}^{|\mathcal{M}^{obs}(t)|}$, one obtains

$$\mathbf{y}^{obs}(t) = \mathbf{O}(t)[\boldsymbol{\pi}(t) + \boldsymbol{\epsilon}(t)] \tag{3}$$

where $\boldsymbol{\epsilon}(t) := [\epsilon_1(t), \ldots, \epsilon_M(t)]^{\mathcal{T}}$ represents the measurement noise, and $\mathbf{O}(t)$ is the matrix consisting of the rows from an $M \times M$ identity matrix corresponding to the CR nodes $m \in \mathcal{M}^{obs}(t)$. Thus, the spectrum sensing problem is to estimate $\boldsymbol{\pi}(t)$ for interpolation (which contains $\pi_m(t)$ for $m \in \mathcal{M}^{miss}(t) := \mathcal{M} \backslash \mathcal{M}^{obs}(t)$) and $\boldsymbol{\pi}(t+1)$ for prediction, given the available measurements $\mathbf{y}^{obs}(t), \mathbf{y}^{obs}(t-1), \ldots, \mathbf{y}^{obs}(1)$.

Note that estimating $\mathbf{G}(t)$ and $\mathbf{p}(t)$ with all observations available (i.e., $\mathcal{M}^{obs}(t) = \mathcal{M}\ \forall t$) was addressed in [8]. A graph Laplacian-based regularizer was shown to be effective in [7] when the number of missing observations is large, and in particular, when the entire observations are missing for some CR nodes m (i.e., $m \in \mathcal{M}^{miss}(t)$ for all t).

III. Dictionary Learning for Dynamic Signals

Motivated by (2), we adopt a bilinear model to represent $\boldsymbol{\pi}(t)$ as a linear combination of a few bases taken possibly from an overcomplete dictionary. Let $\mathbf{D}_1 \in \mathbb{R}^{M \times Q}$ denote a dictionary with Q *atoms*, or bases, that can represent signal $\boldsymbol{\pi}(t)$ with a sparse set of expansion coefficients $\mathbf{s}(t) \in \mathbb{R}^Q$. This amounts to postulating a model

$$\boldsymbol{\pi}(t) = \mathbf{D}_1\mathbf{s}(t) + \mathbf{z}(t) \tag{4}$$

where the nonzero entries in $\mathbf{s}$ are much fewer than Q. Fourier or the wavelet bases are some of the examples that admit such a model for a variety of natural and man-made signals.

Furthermore, to effectively learn and predict spatio-temporal dynamics in a sparsity-leveraging framework, the temporal evolution of $\mathbf{s}(t)$ is also modeled. Again adopting a bilinear model to account for the dynamics that have yet to be learned, using a dictionary $\mathbf{D}_2 \in \mathbb{R}^{Q \times Q}$, we have

$$\mathbf{s}(t) = \mathbf{D}_2\mathbf{s}(t-1) + \mathbf{v}(t). \tag{5}$$

It is clear that (4) and (5) resemble the observation and state equations of a state-space model, except for the prior information regarding the sparsity of $\{\mathbf{s}(t)\}$. That is, if $\mathbf{z}(t)$ and $\mathbf{v}(t)$ were mutually independent and zero-mean Gaussian with covariances $\mathbf{R}$ and $\boldsymbol{\Sigma}$, respectively, $\mathbf{s}(0)$ Guassian with mean $\bar{\mathbf{s}}_0$ and covariance $\boldsymbol{\Sigma}_0$, and $\mathbf{D}_1$ and $\mathbf{D}_2$ were given, the MMSE- and MAP-optimal estimates of $\{\mathbf{s}(t)\}$ can be obtained by solving

$$\min_{\{\mathbf{s}(\tau)\}_{\tau=0}^t} \sum_{\tau=1}^t \Big[||\boldsymbol{\pi}(\tau) - \mathbf{D}_1\mathbf{s}(\tau)||^2_{\mathbf{R}^{-1}} + ||\mathbf{s}(\tau) - \mathbf{D}_2\mathbf{s}(\tau-1)||^2_{\boldsymbol{\Sigma}^{-1}}\Big] + ||\mathbf{s}(0) - \bar{\mathbf{s}}_0||^2_{\boldsymbol{\Sigma}_0^{-1}} \tag{6}$$

where $||\mathbf{x}||^2_{\mathbf{M}} := \mathbf{x}^T\mathbf{M}\mathbf{x}$ for a positive-definite matrix $\mathbf{M}$. The same solution is optimal also for the class of linear estimators even when $\mathbf{z}(t)$, $\mathbf{v}(t)$ and $\mathbf{s}(0)$ are non-Gaussian.

Inspired by this, a Kalman smoother incorporating sparsity of states was proposed in [9]. Here, to jointly learn the observation and the state models in a dictionary learning framework, and also to account for the missing observations per (3), we consider

$$\min_{\mathbf{D}_1 \in \mathcal{D}_1, \mathbf{D}_2 \in \mathcal{D}_2, \{\mathbf{s}(\tau)\}} \sum_{\tau=1}^t \Big[\frac{1}{2}||\mathbf{y}^{obs}(\tau) - \mathbf{O}(\tau)\mathbf{D}_1\mathbf{s}(\tau)||_2^2 + \frac{\mu}{2}||\mathbf{s}(\tau) - \mathbf{D}_2\mathbf{s}(\tau-1)||_2^2 + \lambda||\mathbf{s}(\tau)||_1\Big] \tag{7}$$

where $\mathbf{s}(0) := \bar{\mathbf{s}}_0$,

$$\mathcal{D}_1 := \{[\mathbf{d}_1, \ldots, \mathbf{d}_Q] \in \mathbb{R}^{M \times Q} : ||\mathbf{d}_q||_2^2 \le 1,\ q = 1, \ldots, Q\} \tag{8}$$

and $\mathcal{D}_2$ is similarly defined for Q-by-Q dictionaries. Parameters $\mu \ge 0$ and $\lambda \ge 0$ weight the state evolution fitting term and the sparsity-promoting penalty term, respectively.

Problem (7) is nonconvex, and thus difficult to solve for the global optimum. However, a locally optimal solution can be sought by alternating minimization, based on the fact that with $\mathbf{D}_1$ and $\mathbf{D}_2$ fixed, minimization with respect to (w.r.t.) $\{\mathbf{s}(t)\}$ entails convex optimization, and vice versa.

Therefore, a *batch* algorithm for spectrum sensing can be implemented in two stages. First, in the training stage, given a training set $\{\mathbf{y}_n^{obs}\}_{n=1}^N$, the estimated dictionaries $\hat{\mathbf{D}}_1$ and $\hat{\mathbf{D}}_2$ are obtained by solving

$$(\hat{\mathbf{D}}_1, \hat{\mathbf{D}}_2) = \arg \min_{\mathbf{D}_1 \in \mathcal{D}_1, \mathbf{D}_2 \in \mathcal{D}_2, \{\mathbf{s}_n\}} \sum_{n=1}^N \Big[\frac{1}{2}||\mathbf{y}_n^{obs} - \mathbf{O}_n\mathbf{D}_1\mathbf{s}_n||_2^2 + \frac{\mu}{2}||\mathbf{s}_n - \mathbf{D}_2\mathbf{s}_{n-1}||_2^2 + \lambda||\mathbf{s}_n||_1\Big]. \tag{9}$$

In the operational stage, a sparse state $\hat{\mathbf{s}}(t)$ is estimated at each time $t = 1, 2, \ldots,$ via

$$\hat{\mathbf{s}}(t) = \arg \min_{\{\mathbf{s}(\tau)\}_{\tau=1}^t} \sum_{\tau=1}^t \Big[\frac{1}{2}||\mathbf{y}^{obs}(t) - \mathbf{O}(t)\hat{\mathbf{D}}_1\mathbf{s}(t)||_2^2 + \frac{\mu}{2}||\mathbf{s}(\tau) - \hat{\mathbf{D}}_2\mathbf{s}(\tau-1)||_2^2 + \lambda||\mathbf{s}(\tau)||_1\Big]. \tag{10}$$

Once $\hat{\mathbf{s}}(t)$ is obtained, the desired interference level reconstruction can be found as $\hat{\boldsymbol{\pi}}(t) = \hat{\mathbf{D}}_1\hat{\mathbf{s}}(t)$ and prediction can be carried out as $\hat{\boldsymbol{\pi}}^{pre}(t) = \hat{\mathbf{D}}_1\hat{\mathbf{D}}_2\hat{\mathbf{s}}(t)$.

The batch algorithm becomes computationally very intensive as t grows, since the entire sparse coefficient sequence $\{\mathbf{s}(\tau)\}_{\tau=1}^t$ must be updated at each time t in the operational stage, although an iterative recursive implementation is available [9]. Unfortunately, unlike the original Kalman filtering problem, an efficient *online* implementation does not seem feasible for (10) without sacrificing optimality. Furthermore, the training stage may add up to sensing delay, especially when periodic re-training is necessary due to continual evolution of the dynamic model. In the next section, approximate online implementations are considered.

IV. Online algorithms

As pointed out in Sec. III, an online implementation is desirable, because one does not need to update the whole sequence $\{\mathbf{s}(\tau)\}_{\tau=1}^t$ per time t, and possibly track slow changes in the dictionaries $\mathbf{D}_1$ and $\mathbf{D}_2$. To capture slowly varying dynamics, one can adopt a time-weighted fitting objective, and solve [cf. (7)]

$$\min_{\mathbf{D}_1,\mathbf{D}_2,\{\mathbf{s}(\tau)\}} \sum_{\tau=1}^{t} \beta^{t-\tau} \Big[\frac{1}{2}||\mathbf{y}^{obs}(\tau) - \mathbf{O}(\tau)\mathbf{D}_1\mathbf{s}(\tau)||_2^2 + \frac{\mu}{2}||\mathbf{s}(\tau) - \mathbf{D}_2\mathbf{s}(\tau-1)||_2^2 + \lambda||\mathbf{s}(\tau)||_1\Big] \quad (11)$$

where $0 < \beta \leq 1$ represents a forgetting factor. Instead of solving this in a batch fashion, the idea of online algorithms is to update only the current estimate $\hat{\mathbf{s}}(t)$ at any time t.

A. Online Update for Sparse Coefficients

Suppose first that estimates of dictionaries $\hat{\mathbf{D}}_1(t-1)$ and $\hat{\mathbf{D}}_2(t-1)$ from time $t-1$ are available. The online update for sparse coefficients $\hat{\mathbf{s}}(t)$ has attracted much attention recently [10]–[13]. Given $\hat{\mathbf{s}}(t-1)$, a simple approach is to propagate only the sparse coefficient estimates by solving [12]

$$\hat{\mathbf{s}}(t) = \hat{\mathbf{s}}^{(0)}(t) := \arg\min_{\mathbf{s}} \frac{1}{2}||\mathbf{y}^{obs}(t) - \mathbf{O}(t)\hat{\mathbf{D}}_1(t-1)\mathbf{s}||_2^2 + \frac{\mu}{2}||\mathbf{s} - \hat{\mathbf{D}}_2(t-1)\hat{\mathbf{s}}(t-1)||_2^2 + \lambda||\mathbf{s}||_1. \quad (12)$$

Recall that in the case of Kalman filtering, not only the state estimates, but also the covariance estimates are propagated over time. On the other hand, roughly speaking, the above approach propagates only the first-order statistic, thus significantly undermining optimality [13].

A simple remedy is to update a sliding window of coefficients to trade-off computational complexity for (sub)optimality [14]. With a sliding window of size ℓ, the estimate $\hat{\mathbf{s}}(t)$ (or $\hat{\mathbf{s}}^{(\ell)}(t)$) is obtained as the optimal $\mathbf{s}_0$ of the following optimization problem:

$$\min_{\mathbf{s}_0,\ldots,\mathbf{s}_{-\ell}} \sum_{\tau=t-\ell}^{t} \beta^{t-\tau} \Big[\frac{1}{2}||\mathbf{y}^{obs}(\tau) - \mathbf{O}(\tau)\hat{\mathbf{D}}_1(t-1)\mathbf{s}_{\tau-t}||_2^2 + \frac{\mu}{2}||\mathbf{s}_{\tau-t} - \hat{\mathbf{D}}_2(t-1)\mathbf{s}_{\tau-t-1}||_2^2 + \lambda||\mathbf{s}_{\tau-t}||_1\Big] \quad (13)$$

where $\mathbf{s}_{-(\ell+1)} := \hat{\mathbf{s}}(t-\ell-1)$. Clearly, setting $\ell = t-1$ (and $\beta = 1$) falls back to a batch implementation (cf. (10)), while $\ell = 0$ recovers (12).

B. Online Update for Dictionaries

Once $\hat{\mathbf{s}}(t)$ is obtained, the dictionary update amounts to

$$\min_{\mathbf{D}_1\in\mathcal{D}_1,\mathbf{D}_2\in\mathcal{D}_2} \sum_{\tau=1}^{t} \beta^{t-\tau} \big[||\mathbf{y}^{obs}(\tau) - \mathbf{O}(\tau)\mathbf{D}_1\hat{\mathbf{s}}(\tau)||_2^2 + \mu||\hat{\mathbf{s}}(\tau) - \mathbf{D}_2\hat{\mathbf{s}}(\tau-1)||_2^2\big]. \quad (14)$$

Although the solution depends on the entire observation and sparse coding history, a recursive computation reduces the computational complexity and memory requirement.

Note that (14) is decoupled for $\mathbf{D}_1$ and $\mathbf{D}_2$. To obtain $\hat{\mathbf{D}}_1(t)$, the algorithm proposed in [7] can be used, which is adapted to the present setting here for easy reference. The idea is to

Input: $\{\mathbf{y}^{obs}(t)\}$, $\{\mathbf{O}(t)\}$, $\mathbf{D}_{1,ini}$, $\mathbf{D}_{2,ini}$, $\lambda \geq 0$, $\mu \geq 0$, integer $\ell \geq 0$, $\beta \in (0,1]$
Output: $\{\hat{\boldsymbol{\pi}}(t)\}$ and $\{\hat{\boldsymbol{\pi}}^{pre}(t)\}$

Set $\hat{\mathbf{D}}_1(0) = \mathbf{D}_{1,ini}$, $\mathbf{D}_2(0) = \mathbf{D}_{2,ini}$, $\mathbf{A}_m(0) = \mathbf{0}\ \forall m \in \mathcal{M}$, $\mathbf{A}(0) = \mathbf{0}$, and $\mathbf{B}(0) = \tilde{\mathbf{B}}(0) = \mathbf{0}$.
for $t = 1, 2, \ldots$
  Perform sparse coding
  Solve $\mathbf{s}_0^*, \ldots, \mathbf{s}_{-\ell}^* := \arg\min_{\mathbf{s}_0,\mathbf{s}_{-1},\ldots,\mathbf{s}_{-\ell}}$ $\sum_{\tau=t-\ell}^{t} \beta^{t-\tau} \Big[\frac{1}{2}||\mathbf{y}^{obs}(\tau) - \mathbf{O}(\tau)\hat{\mathbf{D}}_1(t-1)\mathbf{s}_{\tau-t}||_2^2 + \frac{\mu}{2}||\mathbf{s}_{\tau-t} - \hat{\mathbf{D}}_2(t-1)\mathbf{s}_{\tau-t-1}||_2^2 + \lambda||\mathbf{s}_{\tau-t}||_1\Big]$
    where $\mathbf{s}_{-(\ell+1)} := \hat{\mathbf{s}}(t-\ell-1)$, and set $\hat{\mathbf{s}}(t) = \mathbf{s}_0^*$.
  Perform prediction
  Output $\hat{\boldsymbol{\pi}}(t) = \hat{\mathbf{D}}_1(t-1)\hat{\mathbf{s}}(t)$
  and $\hat{\boldsymbol{\pi}}^{pre}(t) = \hat{\mathbf{D}}_1(t-1)\hat{\mathbf{D}}_2(t-1)\hat{\mathbf{s}}(t)$.
  Perform dictionary update
  Update $\{\mathbf{A}_m(t)\}$, $\mathbf{B}(t)$, $\mathbf{A}(t)$ and $\tilde{\mathbf{B}}(t)$ by (15)–(16) and (21)–(22)
  Set $[\hat{\mathbf{d}}_{i,1}(t), \ldots, \hat{\mathbf{d}}_{i,Q}(t)] = \hat{\mathbf{D}}_i(t-1)$, $i = 1, 2$
  Repeat
    For $q = 1, 2, \ldots, Q$
      Update $\hat{\mathbf{d}}_{1,q}(t)$ as (19)–(20) and $\hat{\mathbf{d}}_{2,q}(t)$ as (23)–(24).
    Next q
  Until convergence
  Set $\hat{\mathbf{D}}_i(t) = [\hat{\mathbf{d}}_{i,1}(t), \ldots, \hat{\mathbf{d}}_{i,Q}(t)]$, $i = 1, 2$.
Next t

TABLE I
AN ONLINE ALGORITHM FOR SPECTRUM PREDICTION.

use block coordinate descent (BCD) for the columns of $\mathbf{D}_1$. For this, the following quantities are maintained:

$$\mathbf{A}_m(t) := \sum_{\tau=1}^{t} \beta^{t-\tau} \mathbb{1}_{\{m\in\mathcal{M}^{obs}(\tau)\}} \hat{\mathbf{s}}(\tau)\hat{\mathbf{s}}^{\mathcal{T}}(\tau) = \beta\mathbf{A}_m(t-1) + \mathbb{1}_{\{m\in\mathcal{M}^{obs}(t)\}} \hat{\mathbf{s}}(t)\hat{\mathbf{s}}^{\mathcal{T}}(t),\ m \in \mathcal{M} \quad (15)$$

$$\mathbf{B}(t) := \sum_{\tau=1}^{t} \beta^{t-\tau} \mathbf{O}^{\mathcal{T}}(\tau)\mathbf{y}^{obs}(\tau)\hat{\mathbf{s}}^{\mathcal{T}}(\tau) = \beta\mathbf{B}(t-1) + \mathbf{O}^{\mathcal{T}}(t)\mathbf{y}^{obs}(t)\hat{\mathbf{s}}^{\mathcal{T}}(t) \quad (16)$$

where $\mathbb{1}_{\{\cdot\}}$ is an indicator function equal to 1 if the condition inside the braces are satisfied, and 0 otherwise.

Now, let $\hat{s}_j(\tau)$ denote the j-th entry of $\hat{\mathbf{s}}(\tau)$, and $A_{m,jq}(t)$ and $A_{jq}(t)$ the (j,q)-th entry of matrices $\mathbf{A}_m(t)$ and $\mathbf{A}(t)$, respectively. Also, let $\mathbf{b}_j(t)$ represent the j-th column of $\mathbf{B}(t)$. Then, upon defining

$$\boldsymbol{\Phi}_{j,q}(t) := \sum_{\tau=1}^{t} \beta^{t-\tau} \hat{s}_j(\tau)\hat{s}_q(\tau)\mathbf{O}^{\mathcal{T}}(\tau)\mathbf{O}(\tau) \quad (17)$$

$$= \mathrm{diag}([A_{1,jq}(t), A_{2,jq}(t), \ldots, A_{M,jq}(t)]) \quad (18)$$

the column-wise BCD leads to the following update for the j-th column $\hat{\mathbf{d}}_{1,j}(t)$ of $\hat{\mathbf{D}}_1(t)$:

$$\bar{\mathbf{d}}_{1,j} := \boldsymbol{\Phi}_{j,j}^{-1}(t) \Bigg[\mathbf{b}_j(t) - \sum_{q=1,q\neq j}^{Q} \boldsymbol{\Phi}_{j,q}(t)\hat{\mathbf{d}}_{1,q}(t)\Bigg] \quad (19)$$

$$\hat{\mathbf{d}}_{1,j}(t) = \frac{\bar{\mathbf{d}}_{1,j}}{\max\{||\bar{\mathbf{d}}_{1,j}||_2, 1\}}. \quad (20)$$

The update for $\hat{\mathbf{D}}_2(t)$ is relatively simpler [15]. Upon

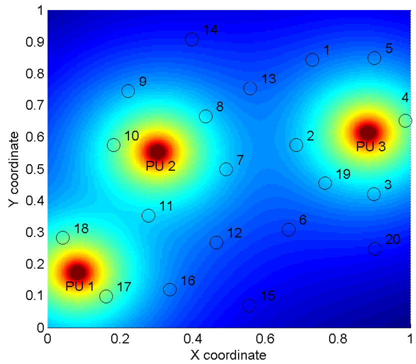

Fig. 1. CR and PU node locations.

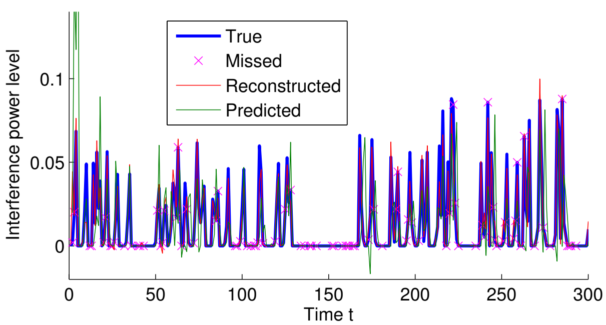

Fig. 2. Reconstructed and predicted interference levels.

defining $\mathbf{A}(t)$ and $\tilde{\mathbf{B}}(t)$ as

$$\mathbf{A}(t) := \beta\mathbf{A}(t-1) + \hat{\mathbf{s}}(t)\hat{\mathbf{s}}^{\mathcal{T}}(t) \tag{21}$$

$$\tilde{\mathbf{B}}(t) := \beta\tilde{\mathbf{B}}(t-1) + \hat{\mathbf{s}}(t)\hat{\mathbf{s}}^{\mathcal{T}}(t-1) \tag{22}$$

the BCD update for the j-th column $\hat{\mathbf{d}}_{2,j}(t)$ of $\hat{\mathbf{D}}_2(t)$ is

$$\bar{\mathbf{d}}_{2,j} := A_{jj}^{-1}(t)\left[\tilde{\mathbf{b}}_j(t) - \hat{\mathbf{D}}_2(t)\mathbf{a}_j\right] + \hat{\mathbf{d}}_{2,j}(t) \tag{23}$$

$$\hat{\mathbf{d}}_{2,j}(t) = \frac{\bar{\mathbf{d}}_{2,j}}{\max\{||\bar{\mathbf{d}}_{2,j}||_2, 1\}} . \tag{24}$$

The overall online algorithm is listed in Table I.

V. Numerical Tests

To assess performance of the proposed algorithms, numerical tests were carried out. A CR network consisting of $M = 20$ nodes with the node locations depicted as circles in Fig. 1 was considered. The interference power distribution due to $K = 3$ PU transmitters is also shown in Fig. 1. The pathloss was set to $(d/d_0)^\alpha$, where d was the distance, $d_0 = 0.01$ and $\alpha = 2.5$. The number of atoms in the dictionaries was set to $Q = 50$. Rayleigh fading channels were simulated using a first-order AR model with coefficient 0.9995. The forgetting factor was set to $\beta = 0.95$. The PU transmit power $p_k(t)$ was drawn independently from a uniform distribution over the interval $[100, 200]$. Temporal correlations in the interference distribution were effected by assuming certain patterns in the PU traffic. At each time t, PU 1 tosses a coin and transmits with probability 0.1. If PU 1 does transmit, then PU 2 will transmit in the next time slot. Likewise, per time t, PU 3 transmits with probability 0.15, followed by PU 2 in the 2nd time slot, and PU 1 in the 3rd. Fig. 2 depicts the reconstructed and predicted interference power levels for CR 1 using $\lambda = 0.005$ and $\mu = 1$ with 30% missing observations employing the online algorithm with $\ell = 10$. It can be seen that the missing observations (marked by crosses) are well reconstructed, and the one-time slot-ahead predictions are quite reliable.

VI. Conclusions

Spatio-temporal dynamics of the interference distribution was learned by a set of cooperating CRs for interpolation of missing observations, as well as for prediction of future interference levels, in a dictionary learning framework. Given the dictionaries, Kalman filter-type formulations with augmented sparsity-promoters were considered. Batch and efficient online solutions have been developed. The numerical tests verified the efficacy of the proposed approach. Experimentation with real datasets will be considered in a future work.

References

[1] S.-J. Kim, E. Dall'Anese, J. A. Bazerque, K. Rajawat, and G. B. Giannakis, "Advances in spectrum sensing and cross-layer design for cognitive radio networks," in *E-Reference Signal Processing*. EURASIP, 2013, in press. [Online]. Available: http://www.dtc.umn.edu/s/resources/eurasip2012tut.pdf

[2] E. Visotsky, S. Kuffner, and R. Peterson, "On collaborative detection of TV transmissions in support of dynamic spectrum sharing," in *Proc. of the DySPAN Conf.*, Baltimore, MD, Nov. 2005, pp. 338–345.

[3] S.-J. Kim, E. Dall'Anese, and G. B. Giannakis, "Cooperative spectrum sensing for cognitive radios using Kriged Kalman filtering," *IEEE J. Sel. Topics Sig. Proc.*, vol. 5, pp. 24–36, Feb. 2011.

[4] S. Yarkan and H. Arslan, "Binary time series approach to spectrum prediction for cognitive radio," in *Proc. of the IEEE Veh. Technol. Conf.*, Baltimore, MD, Oct. 2007, pp. 1563–1567.

[5] X. Li and S. A. Zekavat, "Traffic pattern prediction and performance investigation for cognitive radio systems," in *Proc. of the IEEE Wireless Commun. and Net. Conf.*, Apr. 2008, pp. 894–899.

[6] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 3, pp. 589–600, Apr. 2007.

[7] S.-J. Kim and G. B. Giannakis, "Cognitive radio spectrum prediction using dictionary learning," in *Proc. of the IEEE Global Commun. Conf.*, Atlanta, GA, Dec. 2013.

[8] S.-J. Kim, N. Jain, G. B. Giannakis, and P. Forero, "Joint link learning and cognitive radio sensing," in *Proc. of the 45th Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2011.

[9] D. Angelosante, S. I. Roumeliotis, and G. B. Giannakis, "Lasso-Kalman smoother for tracking sparse signals," in *Proc. of the 43rd Asilomar Conf. on Signals, Syst., and Comput.*, Pacific Grove, CA, Nov. 2009.

[10] A. Carmi, P. Gurfil, and D. Kanevsky, "Methods for sparse signal recovery using Kalman filtering with embedded pseudo-measurement norms and quasi-norms," *IEEE Trans. Sig. Proc.*, vol. 58, no. 4, pp. 2405–2409, Apr. 2010.

[11] N. Vaswani and W. Lu, "Modified-CS: Modifying compressive sensing for problems with partially known support," *IEEE Trans. Sig Proc.*, vol. 58, no. 9, pp. 4595–4607, Sep. 2010.

[12] A. Charles, M. S. Asif, J. Romberg, and C. Rozell, "Sparsity penalties in dynamical system estimation," in *Proc. of the 45th Annual Conf. on Info. Sci. and Syst.*, Baltimore, MD, Mar. 2011.

[13] A. S. Charles and C. J. Rozell, "Re-weighted ℓ_1 dynamic filtering for time-varying sparse signal estimation," Aug. 2012. [Online]. Available: http://arxiv.org/abs/1208.0325

[14] S. Farahmand, G. B. Giannakis, and D. Angelosante, "Doubly robust smoothing of dynamical processes via outlier sparsity constraints," *IEEE Trans. Sig. Proc.*, vol. 59, no. 10, pp. 4529–4543, Oct. 2011.

[15] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, 2010.