Tracking and Smoothing of Time-Varying Sparse Signals via Approximate Belief Propagation

Justin Ziniel, Lee C. Potter, and Philip Schniter

Dept. of ECE, The Ohio State University, Columbus, OH 43210. (Email: {zinielj, potter, schniter}@ece.osu.edu)

Abstract—**This paper considers the problem of recovering time-varying sparse signals from dramatically undersampled measurements. A probabilistic signal model is presented that describes two common traits of time-varying sparse signals: a support set that changes slowly over time, and amplitudes that evolve smoothly in time. An algorithm for recovering signals that exhibit these traits is then described. Built on the belief propagation framework, the algorithm leverages recently developed approximate message passing techniques to perform rapid and accurate estimation. The algorithm is capable of performing both causal tracking and non-causal smoothing to enable both online and offline processing of sparse time series, with a complexity that is linear in all problem dimensions. Simulation results illustrate the performance gains obtained through exploiting the temporal correlation of the time series relative to independent recoveries.**

I. INTRODUCTION

In this work we consider the problem of recovering a time series of sparse signals, $\{\boldsymbol{x}^{(0)}, \boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(T)}\}$, from another time series of undersampled noisy linear measurements, $\{\boldsymbol{y}^{(0)}, \boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(T)}\}$. Such a problem is sometimes referred to in compressive sensing (CS) literature as a *dynamic CS* problem. We assume $\boldsymbol{y}^{(t)} \in \mathbb{C}^M$ is obtained through the linear measurement process

$$\boldsymbol{y}^{(t)} = \boldsymbol{A}^{(t)} \boldsymbol{x}^{(t)} + \boldsymbol{e}^{(t)}, \tag{1}$$

with $\boldsymbol{x}^{(t)} \in \mathbb{C}^N$, and $M < N$. We assume $\boldsymbol{A}^{(t)} \in \mathbb{C}^{M \times N}$ is a complex-valued measurement matrix known in advance, and that its columns have been scaled to be of unit-norm. As in other CS literature, we assume that if the underlying signal being measured is not itself sparse, a sparsifying basis $\boldsymbol{\Psi}$ has been incorporated into $\boldsymbol{A}^{(t)}$ such that $\boldsymbol{x}^{(t)}$ is the sparse representation of the underlying signal in the basis $\boldsymbol{\Psi}$. A complex-valued, stationary random noise process corrupts the measurements, and is modeled as being circular symmetric Gaussian, i.e., $\boldsymbol{e}^{(t)} \sim \mathcal{CN}(\boldsymbol{0}, \sigma_e^2 \boldsymbol{I}_M)$, where $\boldsymbol{I}_M$ denotes an identity matrix of size $M \times M$.

Recently developed methods of solving the dynamic CS problem have been able to outperform traditional CS methods by leveraging one or both of the following empirically observed properties of a large class of time-varying sparse signals: (1) the support sets of the signals change slowly over time, and (2) the amplitudes of the non-zero, or "active", coefficients vary smoothly in time, with few abrupt changes [1]–[5].

In this work, we present a novel approach for recovering time-varying sparse signals that exhibit both of the aforementioned properties. In order to perform principled inference according to our signal model, we develop an iterative algorithm based on the *belief propagation* (BP) methodology [6]. We believe this approach offers several contributions to the dynamic CS literature. First, to the best of our knowledge, ours is the only method which is capable of performing both tracking, in which measurements arrive in a sequential fashion and $\boldsymbol{x}^{(t)}$ is recovered using only $\{\boldsymbol{y}^{(0)}, \ldots, \boldsymbol{y}^{(t)}\}$, and smoothing, in which $\boldsymbol{x}^{(t)}$ is recovered using all available measurements, $\{\boldsymbol{y}^{(t)}\}_{t=0}^T$. This makes our approach suitable for both online and offline processing. Second, our method offers a complexity that is linear in all problem dimensions: $\mathcal{O}(MNT)$. This low complexity burden leads to rapid inference, enabling our method to be applied to large problems while requiring little more than simple matrix-vector products. Third, we offer empirical evidence that our approach is capable of performing near-optimal estimation of signals matched to our signal model by comparing our performance against a genie-aided scheme that bounds the achievable minimum mean square error (MMSE) performance.

II. SIGNAL MODEL

In order to precisely characterize the temporal correlation of the time-varying signals we wish to recover, we adopt a probabilistic signal model, treating the temporal evolution of the signal support as a discrete Markov process, and the temporal variation of coefficient amplitudes as a Gauss-Markov process. We do so by introducing two additional hidden time series that are independent of one another, $\{\boldsymbol{s}^{(t)}\}_{t=0}^T$ and $\{\boldsymbol{\theta}^{(t)}\}_{t=0}^T$. The binary vector $\boldsymbol{s}^{(t)}$ is used to indicate the support of $\boldsymbol{x}^{(t)}$, denoted $\mathcal{S}^{(t)}$, while the complex-valued vector $\boldsymbol{\theta}^{(t)}$ describes the amplitudes of the active coefficients. Together, $\boldsymbol{s}^{(t)}$ and $\boldsymbol{\theta}^{(t)}$ completely characterize $\boldsymbol{x}^{(t)}$ in the following manner:

$$x_n^{(t)} = s_n^{(t)} \cdot \theta_n^{(t)}, \tag{2}$$

where $s_n^{(t)} \in \{0, 1\}$, and $\theta_n^{(t)} \in \mathbb{C}$. Thus, when $s_n^{(t)} = 0$, $x_n^{(t)}$ is inactive, (i.e., $n \notin \mathcal{S}^{(t)}$), and when $s_n^{(t)} = 1$, $x_n^{(t)} = \theta_n^{(t)}$.

To model slow changes in the signal support over time, we treat the time series $\{\boldsymbol{s}^{(t)}\}_{t=0}^T$ as a first-order Markov process in which $s_n^{(t)}$ is conditionally independent of all other support variables, given $s_n^{(t-1)}$, $(t > 0)$. The prior distribution of the supports is fully characterized by two Markov transition probabilities, $p_{10} \triangleq \Pr\{s_n^{(t)} = 1 | s_n^{(t-1)} = 0\}$ and

$p_{01} \triangleq \Pr\{s_n^{(t)} = 0 | s_n^{(t-1)} = 1\}$, and a distribution for the initial timestep. We treat $\{s_n^{(0)}\}_{n=1}^N$ as independent Bernoulli random variables, and define $\overrightarrow{\lambda}_{n0} \triangleq \Pr\{s_n^{(0)} = 1\}$, with $\overrightarrow{\lambda}_{n0}$ chosen small to favor sparse solutions. Depending on how p_{10} and p_{01} are chosen, the prior distribution can favor signals that exhibit a static support across time, or can allow for signal supports that change rapidly over time.

To model the smooth variations of active coefficient amplitudes across time, we adopt a Gauss-Markov process in which the amplitude of each coefficient evolves in time independently from all other coefficients according to

$$\theta_n^{(t)} = (1 - \alpha)\theta_n^{(t-1)} + \alpha w_n^{(t)}, \tag{3}$$

where $w_n^{(t)} \sim \mathcal{CN}(0, \rho)$ is an independent circular Gaussian driving process. The scalar $\alpha \in [0, 1]$ controls the temporal correlation of the active coefficient amplitudes. At one extreme, $\alpha = 0$, the amplitudes remain static across time, i.e., $\theta_n^{(t)} = \theta_n^{(t-1)}$. At the other extreme, $\alpha = 1$, the amplitudes evolve according to an uncorrelated Gaussian random process, and knowledge of a coefficient's amplitude at one timestep provides no information about its amplitude at subsequent timesteps. We model the distribution of amplitudes at the initial timestep as Gaussian, $\boldsymbol{\theta}^{(0)} \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \boldsymbol{I}_N)$.

III. AN ALGORITHM FOR ESTIMATING SPARSE TIME-EVOLVING SIGNALS

In this section we will provide a high level description of an algorithm that efficiently and accurately recovers the unknown time series $\{\boldsymbol{x}^{(t)}\}_{t=0}^T$ from the observed measurements $\{\boldsymbol{y}^{(t)}\}_{t=0}^T$. By defining $\bar{\boldsymbol{y}}$ to be the collection of all measurements, $\{\boldsymbol{y}^{(t)}\}_{t=0}^T$, (and defining $\bar{\boldsymbol{x}}$, $\bar{\boldsymbol{s}}$, and $\bar{\boldsymbol{\theta}}$ similarly), the posterior joint distribution of the signals, supports, and amplitudes, given the time series of measurements, can be expressed using Bayes rule as

$$p(\bar{\boldsymbol{x}}, \bar{\boldsymbol{s}}, \bar{\boldsymbol{\theta}} | \bar{\boldsymbol{y}}) \propto p(\bar{\boldsymbol{y}} | \bar{\boldsymbol{x}}) p(\bar{\boldsymbol{x}} | \bar{\boldsymbol{s}}, \bar{\boldsymbol{\theta}}) p(\bar{\boldsymbol{s}}) p(\bar{\boldsymbol{\theta}}), \tag{4}$$

where $\propto$ indicates proportionality up to a constant scale factor. From (1),

$$p(\bar{\boldsymbol{y}} | \bar{\boldsymbol{x}}) = \prod_{t=0}^T \prod_{m=1}^M \mathcal{CN}(y_m^{(t)}; \boldsymbol{a}_m^{H(t)} \boldsymbol{x}^{(t)}, \sigma_e^2), \tag{5}$$

where $\boldsymbol{a}_m^{H(t)}$ is the mth row of $\boldsymbol{A}^{(t)}$. Furthermore, (2) implies that

$$p(\bar{\boldsymbol{x}} | \bar{\boldsymbol{s}}, \bar{\boldsymbol{\theta}}) = \prod_{t=0}^T \prod_{n=1}^N \delta(x_n^{(t)} - s_n^{(t)} \theta_n^{(t)}), \tag{6}$$

where $\delta(\cdot)$ is the Dirac delta function. Together with our characterization of $p(\bar{\boldsymbol{s}})$ and $p(\bar{\boldsymbol{\theta}})$ from Section II, we see that the terms of (4) and their constituent sub-terms enable the posterior joint pdf to be factored into products of many pdfs that involve small collections of variables. A convenient graphical representation of a joint distribution that decomposes in such a way is given by a *factor graph*. The factor graph for the joint distribution of (4) is shown in Fig. 1. As seen in the figure, all of the variables needed at a given timestep can
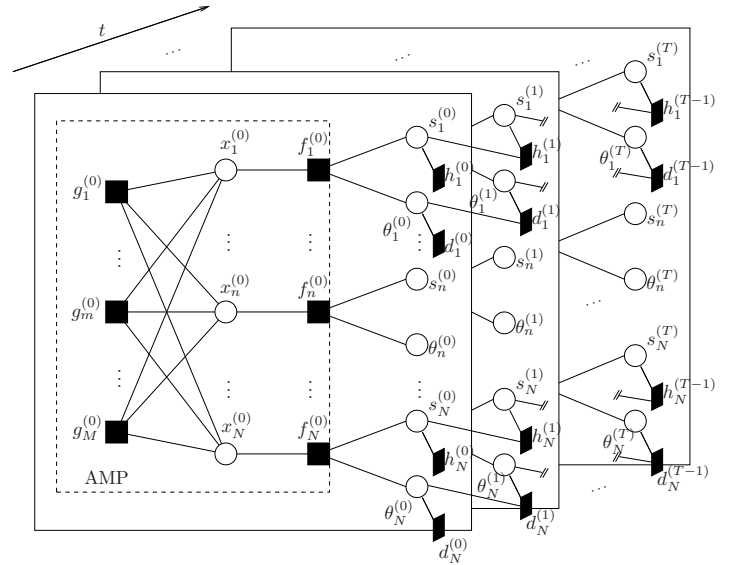

Fig. 1. Factor graph of the time series of sparse signals.

Factor	pdf/pmf
$d_n^{(0)}(\theta_n^{(0)})$	$\mathcal{CN}(\theta_n^{(0)}; 0, \sigma^2)$
$d_n^{(t)}(\theta_n^{(t)}, \theta_n^{(t-1)})$	$\mathcal{CN}(\theta_n^{(t)}; (1-\alpha)\theta_n^{(t-1)}, \alpha^2\rho)$
$g_m^{(t)}(\boldsymbol{x}^{(t)})$	$\mathcal{CN}(y_m^{(t)}; \boldsymbol{a}_m^{H(t)} \boldsymbol{x}^{(t)}, \sigma_e^2)$
$f_n^{(t)}(x_n^{(t)}, s_n^{(t)}, \theta_n^{(t)})$	$\delta(x_n^{(t)} - s_n^{(t)} \theta_n^{(t)})$
$h_n^{(0)}(s_n^{(0)})$	Bernoulli($\overrightarrow{\lambda}_{n0}$)
$h_n^{(t)}(s_n^{(t)}, s_n^{(t-1)})$	Markov $\left(\begin{bmatrix} 1-p_{10} & p_{10} \\ p_{01} & 1-p_{01} \end{bmatrix} \right)$

TABLE I
THE FACTORS AND CORRESPONDING PDFS/PMFS OF FIG. 1

be visualized as lying in a plane, with planes representing consecutive timesteps stacked one after another. We refer to these planes as "frames". The solid square *factor nodes* represent the various pdf (and pmf) "factors" that compose the joint distribution of (4). Each factor node is connected to the circular *variable nodes* upon which the corresponding pdf depends. The specific distributions that correspond to each of the factor nodes are summarized in Table I for both the initial timestep ($t = 0$) and all subsequent timesteps ($t = 1, \ldots, T$).

The temporal correlation of the signal supports is shown by the $h_n^{(t)}$ factor nodes connecting $s_n^{(t)}$ variable nodes between neighboring frames. Likewise, the temporal correlation of the signal amplitudes is expressed by the $d_n^{(t)}$ factor nodes connecting $\theta_n^{(t)}$ variable nodes between neighboring frames. For visual clarity these factor nodes have been omitted from the middle portion of Fig. 1, appearing only at $n = 1$ and $n = N$, but in actuality they are present for all $n = 1, \ldots, N$.

The algorithm we develop is based on the popular belief propagation (BP) approach to performing inference on factor graphs. BP proscribes a set of rules that are used to characterize the pdfs and pmfs, or "messages", that pass between connected nodes of the factor graph [6]. In what follows, we use the notation $\nu_{a \to b}(\cdot)$ to denote a message that is passing from node a to a connected node b. As a brief tutorial on BP, we offer the following two rules that govern how messages are computed: i) an outgoing message from a variable node

along a given edge is the normalized product of the incoming messages to that node from all other edges, and ii) an outgoing message from a factor node along a given edge consists of the normalized integral (or sum) of the product of the local constraint function, (e.g., $d_n^{(t)}$, $f_n^{(t)}$, etc.), and incoming messages on all other connected edges. The normalizations involved ensure that the outgoing pdfs and pmfs are valid, i.e., integrate or sum to one.

A. Message scheduling

There are a number of ways to schedule, or sequence, the messages that are passed through the factor graph. Here, we describe one straightforward method that empirically yields rapid convergence of the signal estimates. For convenience, we will focus only on the messages pertaining to a single coefficient index, n, with the understanding that N such messages would in fact be moving in parallel. Beginning at timestep $t = 0$ we compute $\nu_{s_n^{(0)} \to f_n^{(0)}}(\cdot)$ and $\nu_{\theta_n^{(0)} \to f_n^{(0)}}(\cdot)$, i.e., the messages from $s_n^{(0)}$ and $\theta_n^{(0)}$ that are moving leftward, or further *into* frame 0. These messages in turn depend on the messages originating from the $h_n^{(0)}$ and $d_n^{(0)}$ factor nodes, which can be thought of as providing the priors on $s_n^{(0)}$ and $\theta_n^{(0)}$. Then, the message $\nu_{f_n^{(0)} \to x_n^{(0)}}(\cdot)$ is scheduled, followed by multiple iterations of the messages $\{\nu_{x_n^{(0)} \to g_m^{(0)}}(\cdot)\}_{m=1}^M$ and $\{\nu_{g_m^{(0)} \to x_n^{(0)}}(\cdot)\}_{m=1}^M$. Afterwards, messages begin moving rightward *out* of frame 0, beginning with $\nu_{x_n^{(0)} \to f_n^{(0)}}(\cdot)$, followed by $\nu_{f_n^{(0)} \to s_n^{(0)}}(\cdot)$ and $\nu_{f_n^{(0)} \to \theta_n^{(0)}}(\cdot)$. These latter outbound messages can be thought of as yielding updates to the posterior distributions of $s_n^{(0)}$ and $\theta_n^{(0)}$, given the observations $\boldsymbol{y}^{(0)}$. These updated posterior distributions are in turn used to set the priors for $s_n^{(1)}$ and $\theta_n^{(1)}$ at frame 1 by passing the messages $\nu_{s_n^{(0)} \to h_n^{(1)}}(\cdot)$, $\nu_{h_n^{(1)} \to s_n^{(1)}}(\cdot)$, $\nu_{\theta_n^{(0)} \to d_n^{(1)}}(\cdot)$, and $\nu_{d_n^{(1)} \to \theta_n^{(1)}}(\cdot)$. Then, just as before, messages move from $s_n^{(1)}$ and $\theta_n^{(1)}$ leftward further into frame 1. Subsequently, messages moving out of frame 1 are used to set the priors on $s_n^{(2)}$ and $\theta_n^{(2)}$. Continuing in this manner, messages are propagated forward in time until the last timestep, $t = T$, is reached.

If one wishes to perform tracking, the algorithm can be terminated at this point, as until now it has only made use of sequentially arriving measurements to obtain causal estimates of $\{\boldsymbol{x}^{(t)}\}_{t=0}^T$. If instead one wishes to perform smoothing, using all available measurements to guide the reconstruction of each $\boldsymbol{x}^{(t)}$, one continues as follows. Outgoing messages from frame T are now propagated *backwards* in time, i.e., the messages $\nu_{s_n^{(T)} \to h_n^{(T)}}(\cdot)$, $\nu_{h_n^{(T)} \to s_n^{(T-1)}}(\cdot)$, $\nu_{\theta_n^{(T)} \to d_n^{(T)}}(\cdot)$, and $\nu_{d_n^{(T)} \to \theta_n^{(T-1)}}(\cdot)$ are transmitted. Messages are then passed into and out of the frames in a descending order, updating the priors from the antecedent timesteps until frame 0 is reached. At this point we have completed what we term a single forward/backward pass. Multiple forward/backward passes, indexed by the variable k, can be carried out until convergence in the estimates $\{\hat{\boldsymbol{x}}^{(0)}, \ldots, \hat{\boldsymbol{x}}^{(T)}\}^k$ of the time series occurs, thereby implementing a smoother.

B. Implementation of the message passes

Space constraints prevent us from providing a detailed derivation of the various messages that must be passed in the factor graph of Fig. 1, however some high-level remarks are in order. First, we point out that our choice of $p(\bar{\boldsymbol{s}})$ and $p(\bar{\boldsymbol{\theta}})$ in Section II result in a Bernoulli-Gaussian prior on $x_n^{(t)}$:

$$p(x_n^{(t)}) = (1 - \gamma_n^{(t)})\delta(x_n^{(t)}) + \gamma_n^{(t)}\mathcal{CN}(x_n^{(t)}; 0, \tau_n^{(t)}), \quad (7)$$

where $\gamma_n^{(t)}$ and $\tau_n^{(t)}$ are determined from the signal model parameters. If we were to follow the standard BP rules to compute the messages that pass between the $\{x_n^{(t)}\}$ and $\{g_m^{(t)}\}$ nodes, we would need to evaluate numerous multidimensional integrals at every iteration due to the dense number of interconnections between these nodes. Even more onerous, the number of integrals that would need to be evaluated grows exponentially in M and N.

In order to circumvent this difficulty, we turn to a recently developed technique known as *approximate message passing* (AMP) [7]. AMP, like belief propagation, is not a single algorithm, but rather a framework for constructing algorithms tailored to specific problem setups. The technique leverages two important assumptions in order to drastically simplify the resultant message passing algorithm. First, the message $\nu_{g_m^{(t)} \to x_n^{(t)}}(\cdot)$, which is a function of the product of many non-Gaussian messages, can be well approximated as a Gaussian pdf by applying central limit theorem arguments in the large system limit (i.e., $M, N \to \infty$ with M/N fixed). Approximating this message as a Gaussian bypasses the need to evaluate multidimensional integrals, and allows one to simply track the means and variances of the messages $\{\nu_{x_n^{(t)} \to g_m^{(t)}}(\cdot)\}_{n=1}^N$. The second key assumption, which draws on ideas from the field of statistical physics, is that the many messages leaving a $g_m^{(t)}$ factor node are nearly identical, and can be approximated as such, provided a small but important "correction" term is included in the message computation. As a result of these two assumptions, AMP is able to reduce the number of messages that must be tracked from $\mathcal{O}(MN)$ to $\mathcal{O}(N)$, and the complexity of computing the messages reduces to $\mathcal{O}(MN)$ multiply-and-accumulate operations at each timestep, in the form of matrix-vector products. Additionally, it has recently been shown that AMP enjoys certain theoretical guarantees as well, providing exact posterior distributions in the large system limit, for random Gaussian sensing matrices [8].

We enclosed in a dashed box in Fig. 1 those portions of the factor graph in which we take advantage of AMP approximations of the true BP messages. The AMP messages we use can be shown to be generalizations of the messages derived in [9], which also considered a Bernoulli-Gaussian prior. Using AMP, we are able to characterize all of our messages using a collection of scalar variables, e.g., means, variances, and activity probabilities, which suffice for parameterizing complete distributions. In Fig. 2 we provide a summary of the various messages that pertain to a particular coefficient index n at an intermediate timestep t, as well as the notation that we use for the scalar variables that parameterize the messages.
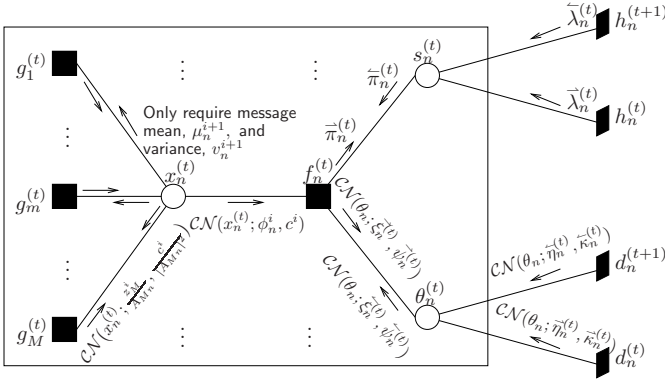
Fig. 2. A summary of the notation and form of messages moving through the factor graph of Fig. 1.

Large arrows indicate the direction that messages are moving. Certain edges only show arrows moving in one direction. In such cases, the messages along "missing" directions can be found implicitly using the provided quantities. Finally, in Table II, we provide pseudo-code for implementing either the tracking version of our algorithm, or the forward portion of a forward/backward smoothing pass.

IV. EMPIRICAL STUDY

As an illustration of the performance of our proposed algorithm, we begin by exhibiting the result of recovering a time-varying sparse signal using three different recovery methods. As a means of obtaining a "gold standard" recovery, we implemented a support-aware Kalman smoother (SKS)[1] which, given perfect knowledge of the time-varying support of the true signal, provides an optimal MMSE estimate of the time series [10]. In order to implement the appropriate Kalman smoother for our signal model, we leveraged the fact that Kalman filtering/smoothing can be viewed as Gaussian message passing on a factor graph that describes a linear state-space model [10].

In addition to the SKS and our proposed method, we implemented an approximate BP algorithm that performs independent recoveries at each timestep (termed "independent AMP"), as well as a timestep-independent Lasso recovery algorithm with genie-aided parameter tuning. Implementing these two additional schemes that do not take advantage of temporal correlation enables us to quantify the improvement gains possible by exploiting correlation in the time series. In Fig. 3, we plot the real part of a single coefficient across $T = 50$ timesteps, along with the real part of the recovery for each of the three methods under consideration. The signal model parameters for the test were $N = 256$, $M = 32$, $\overrightarrow{\lambda}_{n0} = 0.10 \forall n$, $p_{01} = 0.05$ $\alpha = 0.01$, $\sigma^2 = 1$, and a noise variance, σ_e^2, chosen to yield an average per-measurement signal-to-noise ratio (SNR) of 15 dB. The entries of $\boldsymbol{A}^{(t)}$ were drawn i.i.d. from $\mathcal{CN}(0,1)$ and then scaled to yield unit-norm columns. The remaining parameters, p_{10} and ρ, were

[1]To the best of our knowledge, an SKS has previously been developed only for the special case of time-invariant supports, (see [4]), making its extension to the time-varying support case a useful contribution to the literature in its own rite.

% Define important thresholding functions:

$$F_n(\phi; c) \triangleq (1 + \gamma_n(\phi; c))^{-1} \left(\frac{\overleftarrow{\psi}_n^{(t)} \phi + \overleftarrow{\xi}_n^{(t)} c}{\overleftarrow{\psi}_n^{(t)} + c} \right) \quad \text{(D1)}$$

$$G_n(\phi; c) \triangleq (1 + \gamma_n(\phi; c))^{-1} \left(\frac{\overleftarrow{\psi}_n^{(t)} c}{\overleftarrow{\psi}_n^{(t)} + c} \right) + \gamma_n(\phi; c) |F_n(\phi; c)|^2 \quad \text{(D2)}$$

$$F'_n(\phi; c) = \frac{1}{c} G_n(\phi; c) \quad \text{(D3)}$$

$$\gamma_n(\phi; c) \triangleq \left(\frac{1 - \overleftarrow{\pi}_n^{(t)}}{\overleftarrow{\pi}_n^{(t)}} \right) \left(\frac{\overleftarrow{\psi}_n^{(t)} + c}{c} \right)$$
$$\times \exp \left(- \left[\frac{\overleftarrow{\psi}_n^{(t)} |\phi|^2 + \overleftarrow{\xi}_n^{(t)*} c\phi + \overleftarrow{\xi}_n^{(t)} c\phi^* - c|\overleftarrow{\xi}_n^{(t)}|^2}{c(\overleftarrow{\psi}_n^{(t)} + c)} \right] \right) \quad \text{(D4)}$$

% Inputs from previous forward/backward pass:
$$\{\overrightarrow{\lambda}_n^{(0)}\}_{n=1}^N, \{\{\overleftarrow{\lambda}_n^{(t)}\}_{n=1}^N\}_{t=0}^{T-1}, \{\overrightarrow{\eta}_n^{(0)}\}_{n=1}^N,$$
$$\{\{\overleftarrow{\eta}_n^{(t)}\}_{n=1}^N\}_{t=0}^{T-1}, \{\overrightarrow{\kappa}_n^{(0)}\}_{n=1}^N, \{\{\overleftarrow{\kappa}_n^{(t)}\}_{n=1}^N\}_{t=0}^{T-1},$$

% begin passing messages . . .
for $t = 0, \ldots, T, \forall n$:
% pass messages from $s_n^{(t)}$ and $\theta_n^{(t)}$ to $f_n^{(t)}$ . . .

$$\overleftarrow{\pi}_n^{(t)} = \frac{\overrightarrow{\lambda}_n^{(t)} \cdot \overleftarrow{\lambda}_n^{(t)}}{(1 - \overrightarrow{\lambda}_n^{(t)}) \cdot (1 - \overleftarrow{\lambda}_n^{(t)}) + \overrightarrow{\lambda}_n^{(t)} \cdot \overleftarrow{\lambda}_n^{(t)}} \quad \text{(A1)}$$

$$\overleftarrow{\psi}_n^{(t)} = \frac{\overrightarrow{\kappa}_n^{(t)} \cdot \overleftarrow{\kappa}_n^{(t)}}{\overrightarrow{\kappa}_n^{(t)} + \overleftarrow{\kappa}_n^{(t)}} \quad \text{(A2)}$$

$$\overleftarrow{\xi}_n^{(t)} = \overleftarrow{\psi}_n^{(t)} \cdot \left(\frac{\overrightarrow{\eta}_n^{(t)}}{\overrightarrow{\kappa}_n^{(t)}} + \frac{\overleftarrow{\eta}_n^{(t)}}{\overleftarrow{\kappa}_n^{(t)}} \right) \quad \text{(A3)}$$

% initialize AMP-related variables . . .
$\forall m : z_m^1 = y_m^{(t)}, \forall n : \mu_n^1 = 0$, and $c^1 = 100 \cdot \sum_{n=1}^N \psi_n^{(t)}$
% pass I rounds of messages between the $x_n^{(t)}$ and $g_m^{(t)}$ nodes . . .
for $i = 1, \ldots, I, \forall n, m$:

$$\phi_n^i = \sum_{m=1}^M A_{mn}^{*(t)} z_m^i + \mu_n^i \quad \text{(A4)}$$
$$\mu_n^{i+1} = F_n(\phi_n^i; c^i) \quad \text{(A5)}$$
$$v_n^{i+1} = G_n(\phi_n^i; c^i) \quad \text{(A6)}$$
$$c^{i+1} = \sigma_e^2 + \frac{1}{M} \sum_{n=1}^N v_n^{i+1} \quad \text{(A7)}$$
$$z_m^{i+1} = y_m^{(t)} - \boldsymbol{a}_m^{H(t)} \boldsymbol{\mu}^{i+1} + \frac{z_m^i}{M} \sum_{n=1}^N F'_n(\phi_n^i; c^i) \quad \text{(A8)}$$

end
$$\hat{x}_n^{(t)} = \mu_n^{I+1} \qquad \text{\% store current estimate of } x_n^{(t)} \quad \text{(A9)}$$
% now pass messages from $f_n^{(t)}$ out to $s_n^{(t)}$ and $\theta_n^{(t)}$ . . .

$$\overrightarrow{\pi}_n^{(t)} = \left(1 + \left(\frac{\overleftarrow{\pi}_n^{(t)}}{1 - \overleftarrow{\pi}_n^{(t)}} \right) \gamma_n(\phi_n^I; c^{I+1}) \right)^{-1} \quad \text{(A10)}$$

$$\overrightarrow{\xi}_n^{(t)} = \begin{cases} \phi_n^I / \varepsilon, & \overleftarrow{\pi}_n^{(t)} \le 0.99 \\ \phi_n^I, & \text{o.w.} \end{cases} \quad \varepsilon \ll 1 \quad \text{(A11)}$$

$$\overrightarrow{\psi}_n^{(t)} = \begin{cases} c_n^{I+1} / \varepsilon^2, & \overleftarrow{\pi}_n^{(t)} \le 0.99 \\ c_n^{I+1}, & \text{o.w.} \end{cases} \quad \varepsilon \ll 1 \quad \text{(A12)}$$

% pass messages out of frame t, and forward to frame $t+1$ . . .

$$\overrightarrow{\lambda}_n^{(t+1)} = \frac{p_{10}(1 - \overleftarrow{\lambda}_n^{(t)})(1 - \overrightarrow{\pi}_n^{(t)}) + (1 - p_{01})\overrightarrow{\lambda}_n^{(t)}\overrightarrow{\pi}_n^{(t)}}{(1 - \overrightarrow{\lambda}_n^{(t)})(1 - \overrightarrow{\pi}_n^{(t)}) + \overrightarrow{\lambda}_n^{(t)}\overrightarrow{\pi}_n^{(t)}} \quad \text{(A13)}$$

$$\overrightarrow{\eta}_n^{(t+1)} = (1 - \alpha) \left(\frac{\overrightarrow{\kappa}_n^{(t)} \overrightarrow{\psi}_n^{(t)}}{\overrightarrow{\kappa}_n^{(t)} + \overrightarrow{\psi}_n^{(t)}} \right) \left(\frac{\overrightarrow{\eta}_n^{(t)}}{\overrightarrow{\kappa}_n^{(t)}} + \frac{\overrightarrow{\xi}_n^{(t)}}{\overrightarrow{\psi}_n^{(t)}} \right) \quad \text{(A14)}$$

$$\overrightarrow{\kappa}_n^{(t)} = (1 - \alpha)^2 \left(\frac{\overrightarrow{\kappa}_n^{(t)} \overrightarrow{\psi}_n^{(t)}}{\overrightarrow{\kappa}_n^{(t)} + \overrightarrow{\psi}_n^{(t)}} \right) + \alpha^2 \rho \quad \text{(A15)}$$

end

TABLE II
IMPLEMENTATION OF A SINGLE FORWARD PASS

set based on the above parameters such that the number of active coefficients and the variance of the active coefficients would, in expectation, remain fixed across time. Our choice of parameters therefore implies that there are on average only 1.25 measurements-per-active-coefficient. In the realm of traditional CS, this ratio falls well short of the rule-of-thumb of 3-4 measurements-per-active-coefficient, and as evidenced in Fig. 3, the independent AMP and Lasso approaches are unable to accurately estimate the signal. In contrast, both the SKS
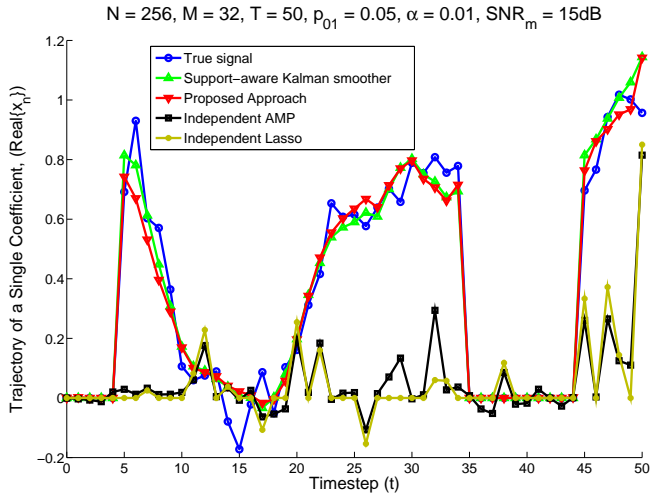
N = 256, M = 32, T = 50, p_{01} = 0.05, α = 0.01, SNR$_m$ = 15dB

Fig. 3.   Sample trajectory of a single coefficient across time.

Fig. 4.   Difference in TNMSE (in dB) between the SKS and our proposed approach (left) and the independent AMP method (right).

and our algorithm are able to accurately track the coefficient's trajectory, even when it transitions between being active and inactive.

To better understand the average performance of our algorithm under a variety of test conditions, we empirically evaluated mean-squared error (MSE) performance on the sparsity-undersampling plane, calculating MSE at various combinations of the normalized sparsity ratio, β (i.e., # of non-zero coefficients to # of measurements), and undersampling ratio, δ (i.e., # of measurements to # of unknowns). In particular, for each pair (δ, β), results were averaged over more than 500 independent signal realizations, and for each realization we computed the time-averaged normalized MSE, TNMSE $\triangleq \frac{1}{T+1} \sum_{t=0}^{T} \frac{\|\boldsymbol{x}^{(t)} - \hat{\boldsymbol{x}}^{(t)}\|_2^2}{\|\boldsymbol{x}^{(t)}\|_2^2}$, where $\hat{\boldsymbol{x}}^{(t)}$ represents a given algorithm's estimate of $\boldsymbol{x}^{(t)}$. Upon computing the realization-averaged TNMSE performance in dB at each (δ, β) pair for the SKS, our proposed approach, and the independent AMP method, we calculated the number of dB by which our method and the independent AMP method exceeded the SKS. In Fig. 4 we have plotted iso-dB contours for a signal model with $N = 512$, a per-measurement SNR of 15 dB, $p_{01} = 0.05$, $\alpha = 0.01$, $(M, \bar{\lambda}_0, p_{10})$ set based on specific (δ, β) pairs, and remaining parameters set as before. To provide an absolute frame of reference, the worst (largest) TNMSE achieved by the SKS over the entire plane was -17 dB, an impressive performance. We do not include comparisons against the independent Lasso approach as we found it performed worse than independent AMP for all of the signals we considered. The advantages of our approach as compared with independent AMP are clearly seen; accounting for the temporal structure of the signal allows us to perform within only a few dB of the SKS over the entire sparsity-undersampling plane. In contrast, independent AMP is more than 10 dB away from the SKS over a substantial portion of the plane.

V. CONCLUSION

In this work we proposed a novel method for recovering time-varying sparse signals that have support sets and amplitudes that evolve slowly over time. We characterized these si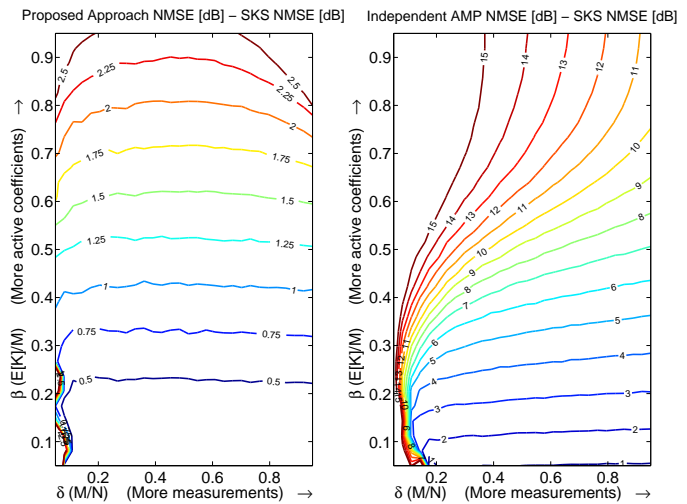gnals within a probabilistic framework, and described a technique for making inferences within this framework that is based on message passing in factor graphs. This technique allows us to rapidly perform both tracking and smoothing with a complexity that is linear in all problem dimensions. In order to have a reference scheme to compare our algorithm's performance against, we implemented a support-aware Kalman smoother for our signal model. Numerical results demonstrated that our method can perform comparably to the Kalman smoother over a much wider range of signal configurations than a traditional CS approach can. Encouraged by these findings, our future efforts will be directed at modeling the additional spatio-temporal structure evident in many time-varying signals, and developing principled methods for learning the parameters of our signal model when they are not perfectly known.

REFERENCES

[1] N. Vaswani, "Kalman filtered compressed sensing," in *IEEE Int'l Conf. on Image Processing (ICIP) 2008*, pp. 893 –896, 12-15 2008.
[2] N. Vaswani and W. Lu, "Modified-CS: Modifying compressive sensing for problems with partially known support," in *IEEE Int'l Symposium on Information Theory (ISIT) 2009*, pp. 488 –492, June 2009.
[3] D. Angelosante, S. Roumeliotis, and G. Giannakis, "Lasso-Kalman smoother for tracking sparse signals," in *Asilomar Conf. on Signals, Systems and Computers 2009*, (Pacific Grove, CA), pp. 181 – 185, Nov. 2009.
[4] D. Angelosante, G. Giannakis, and E. Grossi, "Compressed sensing of time-varying signals," in *Int'l Conf. on Digital Signal Processing 2009*, pp. 1–8, July 2009.
[5] M. Salman Asif, D. Reddy, P. Boufounos, and A. Veeraraghavan, "Streaming compressive sensing for high-speed periodic videos," in *Int'l Conf. on Image Processing (ICIP) 2010*, (Hong Kong), Sept. 2010.
[6] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. San Mateo, CA: Morgan Kaufman, 1988.
[7] D. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," in *Proceedings of the National Academy of Sciences*, vol. 106, pp. 18914–18919, Nov. 2009.
[8] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," in *IEEE Int'l Symposium on Information Theory (ISIT) 2010*, (Austin, TX), June 2010.
[9] P. Schniter, "Turbo reconstruction of structured sparse signals," in *Conf. on Information Sciences and Systems (CISS)*, pp. 1 –6, Mar. 2010.
[10] H. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, and F. Kschischang, "The factor graph approach to model-based signal processing," *Proceedings of the IEEE*, vol. 95, no. 6, pp. 1295–1322, 2007.