Tracking Dynamic Sparse Signals with Kalman Filters: Framework and Improved Inference

Evripidis Karseras , Kin Leung and Wei Dai
Department of Electrical and Electronic Engineering, Imperial College, London, UK
{e.karseras11, kin.leung, wei.dai1}@imperial.ac.uk

***Abstract*—The standard Kalman filter performs optimally for conventional signals but tends to fail when it comes to recovering dynamic sparse signals. In this paper a method to solve this problem is proposed. The basic idea is to model the system dynamics with a hierarchical Bayesian network which successfully captures the inherent sparsity of the data, in contrast to the traditional state-space model. This probabilistic model provides all the necessary statistical information needed to perform sparsity-aware predictions and updates in the Kalman filter steps. A set of theorems show that a properly scaled version of the associated cost function can lead to less greedy optimisation algorithms, unlike the ones previously proposed. It is demonstrated empirically that the proposed method outperforms the traditional Kalman filter for dynamic sparse signals and also how the redesigned inference algorithm, termed here Bayesian Subspace Pursuit (BSP) greatly improves the inference procedure.**

I. Introduction

The Kalman filter has been the workhorse approach in the area of linear dynamic system modelling in both practical and theoretic scenarios. The escalating trend towards sparse signal representation has rendered this estimator to be useless when it comes to tracking *dynamic sparse signals*. It is easy to verify that the estimation process behind the Kalman filter is not fit for sparse signals. Intuitively, the Gaussian *prior* distribution placed over the system's observations does not place any sparsity constraints over the space of all possible solutions.

The Kalman filter was externally modified in the bibliography to admit sparse solutions. The idea in [1] and [2] is to enforce sparsity by thresholds. Work in [3] adopts a probabilistic model but signal amplitudes and support are estimated separately. Finally, the techniques presented in [4] use prior sparsity knowledge into the tracking process. All these approaches typically require a number of parameters to be pre-determined. It also remains unclear how these methods perform towards model and parameter mismatch.

For a single time instance of the sparse reconstruction problem, the Relevance Vector Machine (RVM) introduced in [10] was used with great success in Compressed Sensing applications [5] and basis selection [6]. The hierarchical Bayesian network behind the RVM achieves highly sparse models for the observations not only providing estimates for sparse signals but on their full posterior distributions as well. This is of great importance since it provides all the necessary statistical information to use in the prediction step of the tracking process. Additionally, the inference procedure used in this framework allows for automatic determination of the active components hence the need for a pre-determined level of sparsity is eliminated. This is an appealing attribute for an on-line tracking algorithm.

In this work the aforementioned Bayesian network is employed to extend the state-space model adopted in the traditional Kalman filter. This way the problem of modelling sparsity is tackled efficiently. The resulting statistical information from the inference procedure is then incorporated in the Kalman filter steps thus producing sparsity-aware state estimates.

A set of theorems dictate that a proper scaling of the cost function associated with the inference procedure can lead to more efficient inference algorithms. The techniques initially proposed are greedy methods at heart. By scaling the cost function with the noise variance, and by using knowledge gained from well known compressed sensing algorithms, it is possible to redesign these methods to admit better qualities. The gains are two fold. Firstly, the improved inference mechanism bears far better qualities than the one previously proposed. Secondly, the proposed method outperforms the traditional Kalman filter in terms of reconstruction error when it comes to dynamic sparse signals.

In Section II we present the basic idea for amalgamating the Bayesian network of the RVM in the Kalman filter, termed here Hierarchical Bayesian Kalman filter (HB-Kalman). In Section III we present as set of theorems and explain the motivation to improve upon previous techniques. Additionally we provide the steps for a revised inference algorithm based on the Subspace Pursuit (SP) reconstruction algorithm in [8], termed here Bayesian Subspace Pursuit (BSP). In Section IV we demonstrate the performance of the proposed methods in some synthetic scenarios.

II. Hierarchical Bayesian Kalman Filter

The system model is described by the following equations:

$$\boldsymbol{x}_t = \boldsymbol{F}_t \boldsymbol{x}_{t-1} + \boldsymbol{z}_t, \tag{1}$$

$$\boldsymbol{y}_t = \boldsymbol{\Phi}_t \boldsymbol{x}_t + \boldsymbol{n}_t. \tag{2}$$

where vectors $\boldsymbol{x}_t, \boldsymbol{y}_t$ denote the system's state and observation respectively. The state innovation and observation noise processes are modelled by $\boldsymbol{z}_t$ and $\boldsymbol{n}_t$ respectively.

The authors would like to acknowledge the European Commission for funding SmartEN ITN (Grant No. 238726) under the Marie Curie ITN FP7 programme.

We assume that signal $\boldsymbol{x}_t \in \mathbb{R}^n$ is sparse in some domain. which is considered to remain the same at all time instances (e.g the frames of a video are sparse in the wavelet domain). This allows to set the state transition matrix $\boldsymbol{F}_t$ equal to the unitary matrix $\boldsymbol{I}$. Equation (1) becomes:

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} + \boldsymbol{z}_t$$

As in the standard Kalman filter we adopt the Gaussian assumption so that: $p(\boldsymbol{z}_t) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{Z}_t)$, $p(\boldsymbol{n}_t) = \mathcal{N}\left(\boldsymbol{0}, \sigma^2\boldsymbol{I}\right)$; and $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_{t-1}, \boldsymbol{Z}_t)$ and $p\left(\boldsymbol{y}_t|\boldsymbol{x}_t\right) = \mathcal{N}\left(\boldsymbol{\Phi}\boldsymbol{x}_t, \sigma^2\boldsymbol{I}\right)$. At each time instance, the Kalman filter involves the prediction step where the parameters of $p(\boldsymbol{x}_t|\boldsymbol{y}_{t-1})$ are calculated, while the update step evaluates those of $p(\boldsymbol{x}_t|\boldsymbol{y}_t)$. The advantages of the standard Kalman filter include the ability to track the full statistics, and that the mean squared error solution coincides with the maximum posterior solution which has a closed form. The major issue when applying the filter to dynamic sparse signals, is that the solution is typically not sparse. This drawback is due to the fact that in the standard approach, the covariance matrix $\boldsymbol{Z}_t$ is priorly given. Variants of the Kalman filter such as the non-linear Kalman filter also suffer because of the special nature of the of the non-linearities associated with sparse reconstruction.

To alleviate this problem, the key idea behind Sparse Bayesian Learning (SBL) [10] is employed. As opposed to the traditional Kalman filter where the covariance matrix $\boldsymbol{Z}_t$ of $\boldsymbol{z}_t$ is given, here it is assumed that the state innovation process is given by:

$$\boldsymbol{z}_t \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{A}_t^{-1}\right),$$

where $\boldsymbol{A} = \operatorname{diag}(\boldsymbol{\alpha}) = \operatorname{diag}\left([\alpha_1, \cdots, \alpha_n]_t\right)$, and the hyper-parameters α_i are unknown and have to be learned from $\boldsymbol{y}_t$. To see how this promotes a sparse solution, let us drop the subscript t for simplicity. Then it holds that:

$$p(\boldsymbol{x}|\boldsymbol{\alpha}) = \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{A}^{-1}\right) = \prod_{i=1}^{n} \mathcal{N}\left(0, \alpha_i^{-1}\right).$$

By driving $\alpha_i = +\infty$ it means that $p(x_i|\alpha_i) = \mathcal{N}(0, 0)$; hence it is certain that $x_i = 0$. What remains is to find the maximum likelihood solution of $\boldsymbol{\alpha}$ for the given observation vector $\boldsymbol{y}$. The explicit form of the likelihood function $p\left(\boldsymbol{y}|\boldsymbol{\alpha}, \sigma^2\right)$ was derived in [10] and a set of fast algorithms to estimate $\boldsymbol{\alpha}$ and consequently $\boldsymbol{z}$ and $\boldsymbol{x}$ are proposed in [9].

Finally the principles behind the Kalman filter and SBL are put together. Similar to the standard Kalman filter, two steps, prediction and update, need to be performed at each time instance. In the prediction step, one has to evaluate:

$$\boldsymbol{\mu}_{t|t-1} = \boldsymbol{\mu}_{t-1|t-1}, \quad \boldsymbol{\Sigma}_{t|t-1} = \boldsymbol{\Sigma}_{t-1|t-1} + \boldsymbol{A}_t^{-1},$$
$$\boldsymbol{y}_{t|t-1} = \boldsymbol{\Phi}_t\boldsymbol{\mu}_{t|t-1}, \quad \boldsymbol{y}_{e,t} = \boldsymbol{y}_t - \boldsymbol{y}_{t|t-1}.$$

where the notation $t|t-1$ means prediction at time instance t for measurements up to time instance $t-1$. In the update step, one computes:

$$\boldsymbol{K}_t = \boldsymbol{\Sigma}_{t|t-1}\boldsymbol{\Phi}_t^T(\sigma^2\boldsymbol{I} + \boldsymbol{\Phi}_t\boldsymbol{\Sigma}_{t|t-1}\boldsymbol{\Phi}_t^T)^{-1},$$
$$\boldsymbol{\mu}_{t|t} = \boldsymbol{\mu}_{t|t-1} + \boldsymbol{K}_t\boldsymbol{y}_{e,t}, \quad \boldsymbol{\Sigma}_{t|t} = (\boldsymbol{I} - \boldsymbol{K}_t\boldsymbol{\Phi}_t)\boldsymbol{\Sigma}_{t|t-1}.$$

Differently from the standard Kalman filter, one has to perform the additional step of learning the hyper-parameters $\boldsymbol{\alpha}_t$. From Equation (2) we get $\boldsymbol{y}_{e,t} = \boldsymbol{\Phi}_t\boldsymbol{z}_t + \boldsymbol{n}_t$ where a sparse $\boldsymbol{z}_t$ is preferred to produce a sparse $\boldsymbol{x}_t$. Following the analysis in [10] and [9], maximising the likelihood $p(\boldsymbol{y}_t|\boldsymbol{\alpha}_t)$ is equivalent to minimising the following cost function:

$$\mathcal{L}(\boldsymbol{\alpha}_t) = \log|\boldsymbol{\Sigma}_\alpha| + \boldsymbol{y}_{e,t}^T\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{-1}\boldsymbol{y}_{e,t}, \tag{3}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}} = \sigma^2\boldsymbol{I} + \boldsymbol{\Phi}_t\boldsymbol{A}_t^{-1}\boldsymbol{\Phi}_t^T$. The algorithms described in [9] can be applied to estimate $\boldsymbol{\alpha}_t$. Note that the cost function $\mathcal{L}(\boldsymbol{\alpha})$ is not convex. The obtained estimate $\boldsymbol{\alpha}_t$ is generally sub-optimal and details on the estimation of the globally optimal $\boldsymbol{\alpha}_t$ are given in the next section.

III. Bayesian Subspace Pursuit

Here we discuss the performance guarantees for a single time instance of the inference procedure. For convenience, subscript t is dropped and focus is turned to Equation (2) where $\boldsymbol{x}|\boldsymbol{\alpha} \sim \mathcal{N}\left(\boldsymbol{0}, \boldsymbol{A}^{-1}\right)$. This was analysed in [6] for the purpose of Basis Selection. It had also been proven in [6] that a maximally sparse solution of $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}$ attains the global minimum of the cost function. However, the analysis did not specify the conditions to avoid local minima. By contrast, we provide a more refined analysis. Due to space constraints, only the main results are presented.

We follow [6] by driving the noise variance $\sigma^2 \to 0$. The following Theorem specifies the behaviour of the cost function $\mathcal{L}(\boldsymbol{\alpha})$.

Theorem 1. *For any given $\boldsymbol{\alpha}$, define the set $\mathcal{I} \triangleq \{1 \le i \le n : 0 < \alpha_i < \infty\}$. Then it holds that:*

$$\lim_{\sigma^2\to 0} \sigma^2\mathcal{L}(\boldsymbol{\alpha}) = \left\|\boldsymbol{y} - \boldsymbol{\Phi}_{\mathcal{I}}\boldsymbol{\Phi}_{\mathcal{I}}^{\dagger}\boldsymbol{y}\right\|_2^2, \tag{4}$$

where $\boldsymbol{\Phi}_{\mathcal{I}}$ is a sub-matrix of $\boldsymbol{\Phi}$ formed by the columns indexed by $\mathcal{I}$, and $\boldsymbol{\Phi}_{\mathcal{I}}^{\dagger}$ denotes the pseudo-inverse of $\boldsymbol{\Phi}_{\mathcal{I}}$.

Furthermore, if $|\mathcal{I}| < m$ and $\boldsymbol{y} \in \operatorname{span}(\boldsymbol{\Phi}_{\mathcal{I}})$, then $\mathcal{L}(\boldsymbol{\alpha}) \to -\infty$ and $\sigma^2\mathcal{L}(\boldsymbol{\alpha}) \to 0$ as $\sigma^2 \to 0$.

Two observations can be obtained: (a) the scenarios analysed in [6] can be seen as special cases of Theorem 1 where $\mathcal{L}(\boldsymbol{\alpha}) \to -\infty$; and (b) a proper scaling of the cost function gives the squared ℓ_2-norm of the reconstruction error. Reconstruction is then equivalent to recovering a support set that minimises the reconstruction distortion. This principle is the same as the one behind many greedy algorithms such as the OMP [7] and SP [8]. Theorem 1 suggests such connections.

According to Theorem 1 the key quantities concerning the algorithms described in [9] must be scaled by the noise variance. The original formulae can be found in [9] while the revised ones are given below:

$$\sigma^{-2}\boldsymbol{\Sigma}_x = \left(\sigma^2\boldsymbol{A}_{\mathcal{I}} + \boldsymbol{\Phi}_{\mathcal{I}}^T\boldsymbol{\Phi}_{\mathcal{I}}\right)^{-1}, \quad \boldsymbol{\mu}_x = \sigma^{-2}\boldsymbol{\Sigma}_x\boldsymbol{\Phi}_{\mathcal{I}}^T\boldsymbol{y},$$
$$\sigma^2\boldsymbol{C}_{-i}^{-1} = \boldsymbol{I} - \boldsymbol{\Phi}_{\mathcal{I}-i}\left(\sigma^2\boldsymbol{A}_{\mathcal{I}-i} + \boldsymbol{\Phi}_{\mathcal{I}-i}^T\boldsymbol{\Phi}_{\mathcal{I}-i}\right)^{-1}\boldsymbol{\Phi}_{\mathcal{I}-i}^T,$$
$$\bar{s}_i = \sigma^2 s_i = \boldsymbol{\phi}_i^T\left(\sigma^2\boldsymbol{C}_{-i}^{-1}\right)\boldsymbol{\phi}_i, \quad \bar{q}_i = \sigma^2 q_i = \boldsymbol{\phi}_i^T\left(\sigma^2\boldsymbol{C}_{-i}^{-1}\right)\boldsymbol{y}.$$

Subscript $\mathcal{I}$ denotes the set of indices for which $0 < \alpha_i < +\infty$. The notation $\mathcal{I} - i$ means removal of index i from $\mathcal{I}$.

Subsequently formula [9, Equation (20)] for the optimal α_i given all other α_j, $j \neq i$, becomes:

$$\alpha_i = \frac{\bar{s}_i^2}{\bar{q}_i^2 - \sigma^2 \bar{s}_i}$$

Finally the scaled cost function becomes:

$$\bar{\mathcal{L}} = \sigma^2 \mathcal{L} = \sigma^2 \log \left|\sigma^2 \boldsymbol{I} + \boldsymbol{\Phi}_{\mathcal{I}} \boldsymbol{A}_{\mathcal{I}}^{-1} \boldsymbol{\Phi}_{\mathcal{I}}^T\right| + \boldsymbol{y}^T \left(\boldsymbol{I} - \boldsymbol{\Phi}_{\mathcal{I}} (\sigma^2 \boldsymbol{A}_{\mathcal{I}} + \boldsymbol{\Phi}_{\mathcal{I}}^T \boldsymbol{\Phi}_{\mathcal{I}})^{-1} \boldsymbol{\Phi}_{\mathcal{I}}^T\right) \boldsymbol{y}.$$

Let us clarify the importance of using the scaled quantities. Assume that $\sigma^2 = 0$. It is then easy to show that the original formulae in [9] result in poor performance. Scaling the cost function (and consequently these quantities) is necessary when we want to account for a given noise variance. This may seem irrelevant but in many tracking applications the noise floor is assumed to be estimated in some way or provided by the manufacturer for specific devices. The initial work in [10] provides the formula to infer the noise level from the observations. The scaled versions of the aforementioned quantities can still be applied if desired.

We now have a better understanding of the inference procedure but it still remains unclear what the selection criterion for the basis functions should be. In [9] selection is based on the value of α_i which maximises the difference $\Delta\mathcal{L}$ in the likelihood function, while algorithms such as the OMP and SP make decisions on different grounds. The following Theorem sheds some light on this matter.

Theorem 2. *Assume the noiseless setting $\boldsymbol{y} = \boldsymbol{\Phi x}$ where $\boldsymbol{\Phi} \in \mathbb{R}^{m \times n}$ and $\boldsymbol{\phi}_i^T \boldsymbol{\phi}_i = 1$ for all $1 \leq i \leq n$. Furthermore assume that $t = \max \left|\boldsymbol{\phi}_i^T \boldsymbol{\phi}_j\right|$ for $1 \leq i \neq j \leq n$. Then an algorithm similar to the one in [9] based on one of the following criteria recovers all s-sparse signals exactly given the sufficient condition $t < 0.375/s$; (a) the maximum $\sigma^2 \Delta\mathcal{L}$, (b) the maximum x_i or (c) the minimum α_i.*

Theorem 2 is the starting point for redesigning the inference algorithm. Based on the scaled quantities we can rederive the algorithm in [9] termed Fast Marginal Likelihood Maximisation (FMLM). It is possible to have variants with OMP-like performance guarantees based on different criteria as Theorem 2 suggests. Actually the inference algorithm then greatly resembles the OMP; where the basis functions are recovered sequentially with decreasing order of correlation with the residual signal. For brevity we only present the version based on maximising $\boldsymbol{x}_i$ hence the algorithm is termed FMLM-$\boldsymbol{x}_i$. The steps are given in Algorithm 1.

Theorem 3. *Assume that the same conditions hold as in Theorem 2. An algorithm similar to the one in [9] based on the less greedy criterion of maximum $\theta_i = \bar{q}_i$, recovers all s-sparse signals exactly given the sufficient condition $t < 0.5/s$. The algorithm presented in Algorithm 2 recovers all s-sparse signals exactly if matrix $\boldsymbol{\Phi}$ satisfies the RIP with parameter $\delta_{3s} < 0.205$.*

Theorem 3 suggests further improvements to the performance guarantees, to match those of the OMP by altering the optimisation criterion. Also, results from [8] motivate us to extend the FMLM procedure to a less greedy optimisation procedure by borrowing ideas from the SP algorithm. The SP selects a subset of basis functions at each time instance based also on correlation maximisation, but adds a backtracking step so as to retain only the sparse components with the largest magnitudes. The redesigned algorithm termed here Bayesian Subspace Pursuit is described in Algorithm 2.

Algorithm 1 FMLM-$\boldsymbol{x}_i$

Input: $\boldsymbol{\Phi}, \boldsymbol{y}, \sigma^2$
Initialise:
- $\hat{T} = \{$index $i \in [1, n]$ for maximum $|\boldsymbol{\phi}_i^T \boldsymbol{y}|\}$.

Iteration:
- Calculate values of α_i and $[\boldsymbol{\mu}_x]_i$ for $i \in [1, n] \setminus \hat{T}$.
- $T' = \hat{T} \cup \{$index i corresponding to the maximum value of $[\boldsymbol{\mu}_{\boldsymbol{x}}]_i$ for $i \notin \hat{T}\}$.
- Calculate values α_i for $i \in T'$.
- $\tilde{T} = \{i \in T' : 0 < \alpha_i < +\infty\}$.
- If $|\bar{\mathcal{L}}_{\tilde{T}} - \bar{\mathcal{L}}_{\hat{T}}| = 0$ then compute $\sigma^{-2}\boldsymbol{\Sigma}_x$, $\boldsymbol{\mu}_x$ for $\tilde{T}$ and quit. Set $\hat{T} = \tilde{T}$ and continue otherwise.

Output:
- Estimated support set $\tilde{T}$ and sparse signal $\tilde{\boldsymbol{x}}$ with $|\tilde{T}|$ non-zero components, $\tilde{\boldsymbol{x}}_{\tilde{T}} = \boldsymbol{\mu}_x$.
- Estimated covariance matrix $\sigma^{-2}\boldsymbol{\Sigma}_x$.

Algorithm 2 Bayesian Subspace Pursuit

Input: $\boldsymbol{\Phi}, \boldsymbol{y}, \sigma^2$
Initialise:
- $\hat{T} = \{$index $i \in [1, n]$ for minimum $\alpha_i = \frac{1}{|\boldsymbol{\phi}_i^T \boldsymbol{y}|}\}$.

Iteration:
- Store $\alpha_{max} = \arg\max_{i \in \hat{T}} |\alpha_i|$.
- Calculate values α_i and $\theta_i = \bar{q}_i^2 - \bar{s}_i$ for $i \in [1, n]$.
- Calculate values $t_{\theta_i > 0} = |\{i \in [1, n] : \theta_i > 0\}|$ and $t_{\alpha_i \leq a_{max}} = |\{i \in [1, n] : |\alpha_i| \leq a_{max}\}|$.
- If $t_{\theta_i > 0} = 0$ then $s = t_{\alpha_i \leq a_{max}} + 1$ else $s = t_{\theta_i > 0} + t_{\alpha_i \leq a_{max}}$.
- $T' = \hat{T} \cup \{$indices corresponding to s smallest values of α_i for $i \in [1, n]\}$.
- Compute $\sigma^{-2}\boldsymbol{\Sigma}_x$ and $\boldsymbol{\mu}_x$ for T'.
- $\tilde{T} = \{$indices corresponding to s largest non-zero values of $|\boldsymbol{\mu}_x|$ for which $0 < \alpha_i < +\infty\}$.
- If $|\bar{\mathcal{L}}_{\tilde{T}} - \bar{\mathcal{L}}_{\hat{T}}| = 0$ then quit. Otherwise set $\hat{T} = \tilde{T}$ and continue.

Output:
- Estimated support set $\tilde{T}$ and sparse signal $\tilde{\boldsymbol{x}}$ with $|\tilde{T}|$ non-zero components, $\tilde{\boldsymbol{x}}_{\tilde{T}} = \boldsymbol{\mu}_x$.
- Estimated covariance matrix $\sigma^{-2}\boldsymbol{\Sigma}_x$ for $\tilde{T}$.

IV. Empirical Results

A. Single Time Instance

We concentrate on the performance of the algorithms for a single time instance and for $\sigma^2 = 0$. The algorithms under comparison are the FMLM algorithm as originally presented

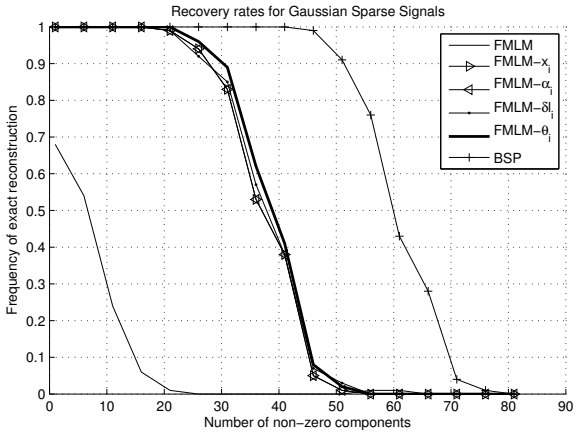

Figure 1. Exact reconstruction rates for $m = 128, n = 256$

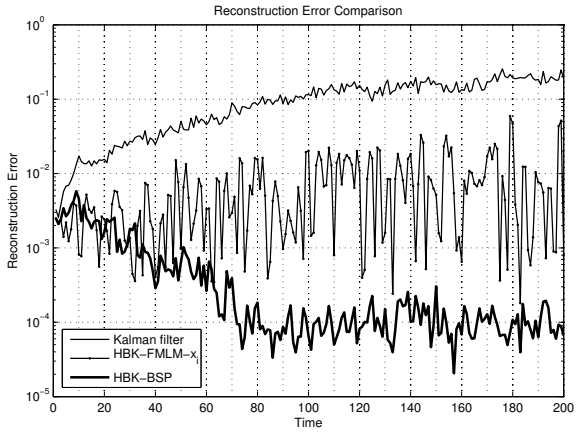

Figure 2. Reconstruction error comparison for $m = 128$.

in [9], the variants based on the scaled quantities; FMLM-$\boldsymbol{x}_i$, FMLM-$\boldsymbol{\alpha}_i$, FMLM-$\boldsymbol{\delta} l_i$, FMLM-$\boldsymbol{\theta}_i$ and the BSP. The experiment is as follows:

1) Generate $\boldsymbol{\Phi} \in \mathbb{R}^{128\times 256}$ with i.i.d entries from $\mathcal{N}\left(0, \frac{1}{m}\right)$.
2) Generate T uniformly at random so that $|T| = K$.
3) Choose values for $\boldsymbol{x}_T$ from $\mathcal{N}(0, 1)$.
4) Compute $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}$ and apply a reconstruction algorithms. Compare estimate $\hat{\boldsymbol{x}}$ to $\boldsymbol{x}$.
5) Repeat experiment for increasing values of K and for 100 realisations.

The results from this procedure are depicted in Figure 1. The first critical observation is that the original FMLM performs poorly when $\sigma^2 = 0$ due to the improperly scaled cost function. The three scaled variants of FMLM based on the criteria mentioned in Theorem 2 perform - within computational accuracy - in the same manner. We observe the increase in performance for FMLM-$\boldsymbol{\theta}_i$, a consequence of altering the selection criterion to $\theta_i = \bar{q}_i$. Even though changing the criterion gives theoretically better performance as Theorem 3 suggests, empirically this gain is not great. By redesigning the inference algorithm based on ideas from the SP we are able to achieve far better performance, as the curve for the BSP algorithm shows.

B. Dynamic Sparse Signal

We now compare the proposed method, HB-Kalman filter against the original Kalman filter. Signal $\boldsymbol{x}_t \in \mathbb{R}^n$ is assumed to be sparse in its natural basis with support set $\mathcal{S}$ chosen uniformly at random from $[1, n]$ where $n = 256$. The magnitudes of the non-zero entries of $\boldsymbol{x}_t$ evolve according to Equation II with $Z_i = \sigma_z^2 I$ with $\sigma_z^2 = 0.1$. The simulation time for this experiment was $T = 200$ time instances. At two randomly chosen time instances: $T = 50$ and $T = 150$, a change in the support of $\boldsymbol{x}_t$ is introduced. A non-zero component is added to the support of $\boldsymbol{x}_{50}$ and a non-zero component is removed from the support of $\boldsymbol{x}_{150}$. Apart from these two time instances the support of $\boldsymbol{x}_t$ remains unchanged. At $T = 1$ the support is initialised with $K = 30$ non-zero components. Observation variance is set to $\sigma_n^2 = 0.01$ for the entire simulation time. We compare the following techniques; the classic Kalman filter, the HBK with FMLM-$\boldsymbol{x}_i$ as the optimisation procedure and the HBK with BSP.

In this scenario noisy measurements $\boldsymbol{y}_t$ are taken by choosing the design matrix $\boldsymbol{\Phi}_t \in \mathbb{R}^{128\times 256}$ as described in subsection IV-A and is re-sampled at each time instance. The number of observations m remains constant at each time instance. In Figure 2 we primarily notice how the HBK outperforms the original Kalman filter, direct consequence of the sparse dynamic model. The HBK-BSP captures the evolution in the support set with greater success due to the improved optimisation algorithm.

REFERENCES

[1] N. Vaswani, "Kalman filtered compressed sensing," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, oct. 2008, pp. 893 –896.

[2] A. Carmi, P. Gurfil, and D. Kanevsky, "Methods for sparse signal recovery using kalman filtering with embedded pseudo-measurement norms and quasi-norms," *Signal Processing, IEEE Transactions on*, vol. 58, no. 4, pp. 2405 –2409, april 2010.

[3] J. Ziniel, L.C. Potter, and P. Schniter, "Tracking and smoothing of time-varying sparse signals via approximate belief propagation," in *Signals, Systems and Computers (ASILOMAR), 2010 Conference Record of the Forty Fourth Asilomar Conference on*. IEEE, 2010, pp. 808–812.

[4] A. Charles, M.S. Asif, J. Romberg, and C. Rozell, "Sparsity penalties in dynamical system estimation," in *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*. IEEE, 2011, pp. 1–6.

[5] Shihao Ji, Ya Xue, and L. Carin, "Bayesian compressive sensing," *Signal Processing, IEEE Transactions on*, vol. 56, no. 6, pp. 2346 –2356, june 2008.

[6] D.P. Wipf and B.D. Rao, "Sparse bayesian learning for basis selection," *Signal Processing, IEEE Transactions on*, vol. 52, no. 8, pp. 2153 – 2164, aug. 2004.

[7] J.A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Trans. Inform. Theory*, vol. 50, no. 10, pp. 2231 – 2242, oct. 2004.

[8] Wei Dai and Olgica Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inform. Theory*, vol. 55, pp. 2230–2249, 2009.

[9] M.E. Tipping, A.C. Faul, et al., "Fast marginal likelihood maximisation for sparse Bayesian models," in *International workshop on artificial intelligence and statistics*. Jan, 2003, vol. 1.

[10] Michael E. Tipping, "The relevance vector machine," 2000.