REGULARIZED STATE ESTIMATION AND PARAMETER LEARNING VIA AUGMENTED LAGRANGIAN KALMAN SMOOTHER METHOD

Rui Gao, Filip Tronarp, Zheng Zhao, Simo Särkkä

Aalto University
Department of Electrical Engineering and Automation
Rakentajanaukio 2C, Espoo, Finland

ABSTRACT

In this article, we address the problem of estimating the state and learning of the parameters in a linear dynamic system with generalized L_1-regularization. Assuming a sparsity prior on the state, the joint state estimation and parameter learning problem is cast as an unconstrained optimization problem. However, when the dimensionality of state or parameters is large, memory requirements and computation of learning algorithms are generally prohibitive. Here, we develop a new augmented Lagrangian Kalman smoother method for solving this problem, where the primal variable update is reformulated as Kalman smoother. The effectiveness of the proposed method for state estimation and parameter learning is demonstrated in spectro-temporal estimation tasks using both synthetic and real data.

Index Terms— state estimation, parameter learning, sparsity, Kalman smoother, augmented Lagrangian method

1. INTRODUCTION

State estimation and parameter learning in linear dynamic systems is an important problem that arises in a wide range of applications such as target tracking, inertial navigation, and biomedical signal analysis [1–3]. The estimation and learning task is generally cast as a statistical inference problem for recovering the original time series and learning the unknown parameters from noisy measurements, based upon reasonable assumptions or prior information about the observations [1, 2].

In recent years, a variety of sparsity-promoting formulations and algorithms for state estimation as well as parameter learning have been considered in literature [4–6]. However, it is known that it is hard to promote sparsity using probabilistic approaches for learning parameters. To alleviate this problem, studies have emerged that attempt to estimate the state or learn parameters by incorporating sparse optimization methods. For example, in synthesis sparse models [4, 5], the basic idea is to represent time series as a linear combination of a limited number of basis coefficients, and to estimate the coefficients to construct sparse approximations or estimators. Then, the parameters are learned by using penalizer terms along with a sparsity-inducing regularizer. However, this approach is restricted to the state being sparse in its original domain. Another approach is based on analysis sparsity, which assumes the state is sparse in some transform domain [7, 8]. While improving the learning process, this approach has computational challenges in large-scale linear dynamic systems.

In this article, we present an efficient method for regularized state estimation and parameter learning from noisy measurements. The main contributions of this article are (1) to build a generalized L_1-regularized model for joint estimation of the state and learning of the parameters, and (2) to propose a new augmented Lagrangian Kalman smoother method to solve this problem. We base our method on the results of our article [9], where the emphasis is on nonlinear system with known parameters. Experimental results demonstrate that the proposed method has low computational complexity and high accuracy, especially in large datasets. We also apply the method to automatic transcription of music from audio signals. The overall architecture is shown in Fig.1.

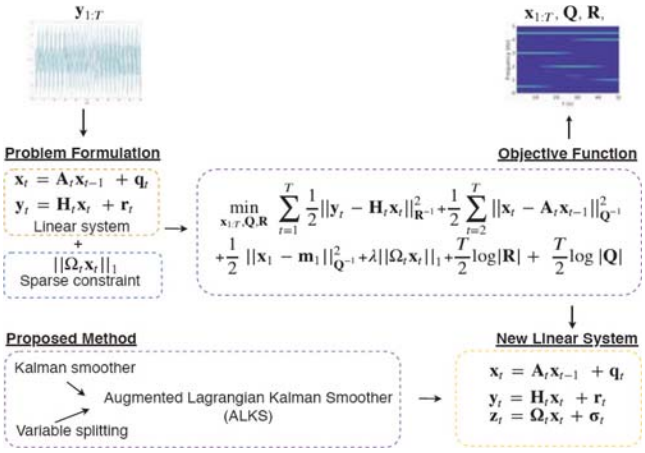

$\mathbf{y}_{1:T}$
$\mathbf{x}_{1:T}$, $\mathbf{Q}$, $\mathbf{R}$,
Problem Formulation
$\mathbf{x}_t = \mathbf{A}_t\mathbf{x}_{t-1} + \mathbf{q}_t$
$\mathbf{y}_t = \mathbf{H}_t\mathbf{x}_t + \mathbf{r}_t$
Linear system
+
$||\Omega_t\mathbf{x}_t||_1$
Sparse constraint
Objective Function
$\min_{\mathbf{x}_{1:T},\mathbf{Q},\mathbf{R}} \sum_{t=1}^{T} \frac{1}{2}||\mathbf{y}_t - \mathbf{H}_t\mathbf{x}_t||^2_{\mathbf{R}^{-1}} + \frac{1}{2}\sum_{t=2}^{T} ||\mathbf{x}_t - \mathbf{A}_t\mathbf{x}_{t-1}||^2_{\mathbf{Q}^{-1}}$
$+\frac{1}{2}||\mathbf{x}_1 - \mathbf{m}_1||^2_{\mathbf{Q}^{-1}} + \lambda||\Omega_t\mathbf{x}_t||_1 + \frac{T}{2}\log|\mathbf{R}| + \frac{T}{2}\log|\mathbf{Q}|$
Proposed Method
Kalman smoother
Variable splitting
Augmented Lagrangian Kalman Smoother
(ALKS)
New Linear System
$\mathbf{x}_t = \mathbf{A}_t\mathbf{x}_{t-1} + \mathbf{q}_t$
$\mathbf{y}_t = \mathbf{H}_t\mathbf{x}_t + \mathbf{r}_t$
$\mathbf{z}_t = \Omega_t\mathbf{x}_t + \sigma_t$

Fig. 1: Our architecture. A generalized L_1-regularized cost function is used to estimate the state and to learn the parameters via an augmented Lagrangian using a Kalman smoother.

978-1-7281-0824-7/19/$31.00 ©2019 IEEE

2. THE MODEL AND PROBLEM FORMULATION

Let $\mathbf{y}_t = \left[y_{1,t}, y_{2,t}, \ldots, y_{n_y,t}\right]^\top \in \mathbb{R}^{n_y}$ be a noisy measurement signal of the system at time step t, $\mathbf{H}_t \in \mathbb{R}^{n_y \times n_x}$ a measurement matrix, and $\mathbf{A}_t \in \mathbb{R}^{n_x \times n_x}$ a state transition matrix. Then, we have the following linear state space model

$$\begin{aligned} \mathbf{x}_t &= \mathbf{A}_t \mathbf{x}_{t-1} + \mathbf{q}_t, \\ \mathbf{y}_t &= \mathbf{H}_t \mathbf{x}_t + \mathbf{r}_t, \qquad t = 1, \ldots, T, \end{aligned} \tag{1}$$

where $\mathbf{x}_t = [x_{1,t}, x_{2,t}, \ldots, x_{n_x,t}]^\top \in \mathbb{R}^{n_x}$ denotes the unknown state of the system at time step t, and $\mathbf{q}_t$ and $\mathbf{r}_t$ are independent zero-mean Gaussian noises with covariances $\mathbf{Q}$, $\mathbf{R}$, respectively. The initial state $\mathbf{x}_1$ is assumed to be Gaussian with mean $\mathbf{m}_1$ and covariance $\mathbf{P}_1$.

Our goal here is to estimate the state sequence $\mathbf{x}_{1:T}$, and covariances $\mathbf{Q}$ and $\mathbf{R}$, given the observations $\mathbf{y}_{1:T}$. In the Bayesian approach, the unknown parameter is assigned a prior distribution, and the posterior density of the parameters given the observations is to be computed. We denote the joint posterior probability density of $\mathbf{x}_{1:T}$, $\mathbf{Q}$, and $\mathbf{R}$ conditioned on the measurements by $p(\mathbf{x}_{1:T}, \mathbf{Q}, \mathbf{R} \mid \mathbf{y}_{1:T})$. By Bayes' rule, we have that

$$\begin{aligned} &p(\mathbf{x}_{1:T}, \mathbf{Q}, \mathbf{R} \mid \mathbf{y}_{1:T}) \\ &\quad \propto p(\mathbf{x}_1) \prod_{t=1}^{T} p(\mathbf{y}_t \mid \mathbf{x}_t) \prod_{t=2}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}), \end{aligned} \tag{2}$$

where $\propto$ denotes proportionality and

$$\begin{aligned} p(\mathbf{y}_t \mid \mathbf{x}_t) &= \mathcal{N}(\mathbf{y}_t \mid \mathbf{H}_t \mathbf{x}_t, \mathbf{R}), \\ p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) &= \mathcal{N}(\mathbf{x}_t \mid \mathbf{A}_t \mathbf{x}_{t-1}, \mathbf{Q}), \end{aligned} \tag{3}$$

where $\mathcal{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P})$ denotes a Gaussian probability density function with mean $\mathbf{m}$ and covariance $\mathbf{P}$ evaluated at $\mathbf{x}$.

Additionally, we assume that $\mathbf{x}_t$ admits a sparse representation in the range of an analysis operator $\mathbf{\Omega}_t \in \mathbb{R}^{P \times n_x}$. This is achieved by adding Laplace distributed pseudo-measurements of $\mathbf{\Omega}_t \mathbf{x}_t$ [7]. Finally, using flat the priors for the covariances $p(\mathbf{R}) \propto 1$ and $p(\mathbf{Q}) \propto 1$, gives the *maximum a posteriori* (MAP) estimate, which is computed by minimizing its negative log posterior

$$\begin{aligned} \{\mathbf{x}^\star_{1:T}, \mathbf{Q}^\star, \mathbf{R}^\star\} &= \operatorname*{arg\,min}_{\mathbf{x}_{1:T}, \mathbf{Q}, \mathbf{R}} \sum_{t=1}^{T} \frac{1}{2} \|\mathbf{y}_t - \mathbf{H}_t \mathbf{x}_t\|^2_{\mathbf{R}^{-1}} \\ &+ \frac{1}{2} \sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{A}_t \mathbf{x}_{t-1}\|^2_{\mathbf{Q}^{-1}} + \frac{1}{2} \|\mathbf{x}_1 - \mathbf{m}_1\|^2_{\mathbf{P}_1^{-1}} \\ &+ \sum_{t=1}^{T} \lambda \|\mathbf{\Omega}_t \mathbf{x}_t\|_1 + \frac{T}{2} \log|\mathbf{R}| + \frac{T}{2} \log|\mathbf{Q}| + \text{constant}. \end{aligned} \tag{4}$$

Here, $\mathbf{x}^\star_{1:T}, \mathbf{Q}^\star, \mathbf{R}^\star$ are the MAP estimates of the parameters, λ is a penalty parameter, and $\|\cdot\|_{\mathbf{R}^{-1}}$ is a $\mathbf{R}^{-1}$-weighted Euclidean norm. Note that if we did not use any penalty terms (i.e., when $\lambda = 0$), the optimization problem could be solved by using the Kalman filter and Rauch–Tung–Striebel (RTS) smoother [2, 6, 10], also called Kalman smoother (KS). However, when $\lambda > 0$, the minimization problem in (4) is complicated in practice and needs to be performed approximately. In the following, we propose a new augmented Lagrangian Kalman smoother to solve this problem.

3. LEARNING THE PARAMETERS

To learn the parameters in (4), a lot of variable splitting optimization algorithms have been proposed [11–14]. Since these methods utilize the matrix-inverse computation at each iteration, such methods are not efficient when the dataset is large. Therefore we need an alternative approach to tackle the large-scale state estimation and parameter learning problem. Our new method builds upon the batch augmented Lagrangian framework [13, 14]. The main idea here is to build the augmented Lagrangian function and to decompose the objective into an iterative sequence of much easier subproblems. Then, the subproblems are solved by applying KS and the proximal operator. The details of the proposed method are described below.

We start with introducing a sequence of auxiliary variables $\mathbf{w}_t$ and rewrite (4) by using variable-splitting as follows

$$\begin{aligned} \min_{\mathbf{x}_{1:T}, \mathbf{Q}, \mathbf{R}} \; & \frac{1}{2} \sum_{t=1}^{T} \|\mathbf{y}_t - \mathbf{H}_t \mathbf{x}_t\|^2_{\mathbf{R}^{-1}} + \frac{1}{2} \|\mathbf{x}_1 - \mathbf{m}_1\|^2_{\mathbf{P}_1^{-1}} \\ & + \frac{1}{2} \sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{A}_t \mathbf{x}_{t-1}\|^2_{\mathbf{Q}^{-1}} + \sum_{t=1}^{T} \lambda \|\mathbf{w}_t\|_1 \\ & + \frac{T}{2} \log|\mathbf{Q}| + \frac{T}{2} \log|\mathbf{R}| \\ \text{s.t.} \quad & \mathbf{w}_t = \mathbf{\Omega}_t \mathbf{x}_t, \quad t = 1, \ldots, T. \end{aligned} \tag{5}$$

The augmented Lagrangian function associated with the problem (4) is given by

$$\begin{aligned} \mathcal{L}(\mathbf{x}_{1:T}, \mathbf{Q}, \mathbf{R}, \mathbf{w}_{1:T}; \boldsymbol{\eta}_{1:T}) &\triangleq \frac{1}{2} \sum_{t=1}^{T} \|\mathbf{y}_t - \mathbf{H}_t \mathbf{x}_t\|^2_{\mathbf{R}^{-1}} \\ &+ \frac{1}{2} \sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{A}_t \mathbf{x}_{t-1}\|^2_{\mathbf{Q}^{-1}} + \frac{1}{2} \|\mathbf{x}_1 - \mathbf{m}_1\|^2_{\mathbf{P}_1^{-1}} \\ &+ \lambda \sum_{t=1}^{T} \|\mathbf{w}_t\|_1 + \frac{T}{2} \log|\mathbf{Q}| + \frac{T}{2} \log|\mathbf{R}| \\ &+ \frac{\rho}{2} \sum_{t=1}^{T} \|\mathbf{\Omega}_t \mathbf{x}_t - \mathbf{w}_t + \boldsymbol{\eta}_t\|^2, \end{aligned} \tag{6}$$

where $\rho > 0$ is a penalty parameter and $\boldsymbol{\eta}_t,\ t = 1, \ldots, T$ are dual variables. Although there are no convergent guarantees when there are more than two blocks or the subproblem is non-convex, the augmented Lagrangian method has been considered in many applications [15]. The values of $\mathbf{x}_{1:T}$, $\mathbf{Q}$, $\mathbf{R}$,

$\mathbf{w}_{1:T}$ and $\boldsymbol{\eta}_{1:T}$ are updated by minimizing $\mathcal{L}$ in an alternating manner. Namely, at each iteration, the updates are

$$\begin{aligned}
\mathbf{x}_{1:T}^{(k+1)} &= \arg\min_{\mathbf{x}_{1:T}} \mathcal{L}(\mathbf{x}_{1:T}, \mathbf{Q}^{(k)}, \mathbf{R}^{(k)}, \mathbf{w}_{1:T}^{(k)}; \boldsymbol{\eta}_{1:T}^{(k)}),\\
\{\mathbf{Q}^{(k+1)}, \mathbf{R}^{(k+1)}\} &= \arg\min_{\mathbf{Q},\mathbf{R}} \mathcal{L}(\mathbf{x}_{1:T}^{(k+1)}, \mathbf{Q}, \mathbf{R}, \mathbf{w}_{1:T}^{(k)}; \boldsymbol{\eta}_{1:T}^{(k)}),\\
\mathbf{w}_{1:T}^{(k+1)} &= \arg\min_{\mathbf{w}_{1:T}} \mathcal{L}(\mathbf{x}_{1:T}^{(k+1)}, \mathbf{Q}^{(k+1)}, \mathbf{R}^{(k+1)}, \mathbf{w}_{1:T}; \boldsymbol{\eta}_{1:T}^{(k)}),\\
\boldsymbol{\eta}_t^{(k+1)} &= \boldsymbol{\eta}_t^{(k)} + (\boldsymbol{\Omega}_t \mathbf{x}_t^{(k+1)} - \mathbf{w}_t^{(k+1)}), \quad t = 1, \ldots, T,
\end{aligned} \tag{7}$$

which is also a special form of scaled alternating direction method of multipliers [16]. The computational complexity of the augmented Lagrangian method is largely dependent on how fast we can solve the $\mathbf{x}_{1:T}$-subproblem. While the $\mathbf{x}_{1:T}$-subproblem can be solved in closed form, a direct batch solution becomes computationally burdensome due to large matrix computations at each iteration. This motivates us to integrate the filter and smoother trick into the augmented Lagrangian method below.

In our specialized augmented Lagrangian method, we utilize KS to solve the $\mathbf{x}_{1:T}$-subproblem. More specifically, we define an artificial measurement noise $\boldsymbol{\sigma}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ and a pseudo-measurement $\mathbf{z}_t$, expressed as (cf. [9])

$$\begin{aligned}
\boldsymbol{\Sigma} &= \mathbf{I}/\rho,\\
\mathbf{z}_t &= \mathbf{w}_t - \boldsymbol{\eta}_t,
\end{aligned} \tag{8}$$

hence writing the $\mathbf{x}_{1:T}$-subproblem in (7) as

$$\begin{aligned}
&\min_{\mathbf{x}_{1:T}} \frac{1}{2}\sum_{t=1}^{T} \|\mathbf{y}_t - \mathbf{H}_t\mathbf{x}_t\|_{\mathbf{R}^{-1}}^2 + \frac{1}{2}\sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{A}_t\mathbf{x}_{t-1}\|_{\mathbf{Q}^{-1}}^2\\
&+ \frac{1}{2}\sum_{t=1}^{T} \|\mathbf{z}_t - \boldsymbol{\Omega}_t\mathbf{x}_t\|_{\boldsymbol{\Sigma}^{-1}}^2 + \frac{1}{2}\|\mathbf{x}_1 - \mathbf{m}_1\|_{\mathbf{P}_1^{-1}}^2.
\end{aligned} \tag{9}$$

The minimization problem in (9) can be solved by KS [2, 17]. For $t = 1, \ldots, T$, the forward recursion of KS is

$$\begin{aligned}
\mathbf{m}_t^- &= \mathbf{A}_t \mathbf{m}_{t-1},\\
\mathbf{P}_t^- &= \mathbf{A}_t \, \mathbf{P}_{t-1} \, \mathbf{A}_t^\top + \mathbf{Q},\\
\mathbf{S}_t^y &= \mathbf{H}_t \, \mathbf{P}_t^- \, \mathbf{H}_t^\top + \mathbf{R},\\
\mathbf{K}_t^y &= \mathbf{P}_t^- \, \mathbf{H}_t^\top \, [\mathbf{S}_t^y]^{-1},\\
\mathbf{m}_t^y &= \mathbf{m}_t^- + \mathbf{K}_t^y \left[\mathbf{y}_t - \mathbf{H}_t \, \mathbf{m}_t^-\right],\\
\mathbf{P}_t^y &= \mathbf{P}_t^- - \mathbf{K}_t^y \, \mathbf{S}_t^y \, [\mathbf{K}_t^y]^\top,\\
\mathbf{S}_t^z &= \boldsymbol{\Omega}_t \, \mathbf{P}_t^y \, \boldsymbol{\Omega}_t^\top + \boldsymbol{\Sigma},\\
\mathbf{K}_t^z &= \mathbf{P}_t^y \, \boldsymbol{\Omega}_t^\top \, [\mathbf{S}_t^z]^{-1},\\
\mathbf{m}_t &= \mathbf{m}_t^y + \mathbf{K}_t^z \, [\mathbf{z}_t - \boldsymbol{\Omega}_\mathbf{t} \mathbf{m}_t^y],\\
\mathbf{P}_t &= \mathbf{P}_t^y - \mathbf{K}_t^z \, \mathbf{S}_t^z \, [\mathbf{K}_t^z]^\top,
\end{aligned} \tag{10}$$

where $\mathbf{m}_t^-$ and $\mathbf{P}_t^-$ are the predicted mean and covariance at the time t. Additionally, $\mathbf{S}_t^y$ and $\mathbf{S}_t^z$, $\mathbf{K}_t^y$ and $\mathbf{K}_t^z$, $\mathbf{m}_t^y$ and $\mathbf{m}_t$, $\mathbf{P}_t^y$ and $\mathbf{P}_t$ are the innovation covariances, gain matrices, means, and covariances for $\mathbf{y}_t$ and $\mathbf{z}_t$ at the time step t, respectively. The backward recursion uses the results above and iterates backwards according to

$$\begin{aligned}
\mathbf{G}_t &= \mathbf{P}_t \, \mathbf{A}_{t+1}^\top \, \left[\mathbf{P}_{t+1}^-\right]^{-1},\\
\mathbf{m}_t^s &= \mathbf{P}_t + \mathbf{G}_t \, \left[\mathbf{m}_{t+1}^s - \mathbf{m}_{t+1}^-\right],\\
\mathbf{P}_t^s &= \mathbf{P}_t + \mathbf{G}_t \, \left[\mathbf{P}_{t+1}^s - \mathbf{P}_{t+1}^-\right] \, \mathbf{G}_t^\top,
\end{aligned} \tag{11}$$

which should be started from the Kalman filter result at time step T (see [2] for details). After the above recursion, we have

$$\mathbf{x}_{1:T}^{(k+1)} = \mathbf{m}_{1:T}^s. \tag{12}$$

The KS approach presented here avoids high memory and computational requirements, and thus is fast and easy to implement. The extra updates of $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{w}_{1:T}$, and $\boldsymbol{\eta}_{1:T}$-subproblems are analogous to those from the standard augmented Lagrangian method. The $\mathbf{Q}$-subproblem and $\mathbf{R}$-subproblem are analytically expressed as

$$\begin{aligned}
&\min_{\mathbf{Q}} \frac{1}{2}\sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{A}_t\mathbf{x}_{t-1}\|_{\mathbf{Q}^{-1}}^2 + \frac{T-1}{2}\log|\mathbf{Q}|,\\
&\min_{\mathbf{R}} \frac{1}{2}\sum_{t=1}^{T} \|\mathbf{y}_t - \mathbf{H}_t\mathbf{x}_t\|_{\mathbf{R}^{-1}}^2 + \frac{T}{2}\log|\mathbf{R}|.
\end{aligned} \tag{13}$$

Then, by setting the derivatives of (13) to zero, we obtain the following expressions

$$\begin{aligned}
\mathbf{Q}^{(k+1)} &= \frac{1}{T-1}\sum_{t=2}^{T} (\mathbf{x}_t - \mathbf{A}_t\mathbf{x}_{t-1})(\mathbf{x}_t - \mathbf{A}_t\mathbf{x}_{t-1})^\top,\\
\mathbf{R}^{(k+1)} &= \frac{1}{T}\sum_{t=1}^{T} (\mathbf{y}_t - \mathbf{H}_t\mathbf{x}_t)(\mathbf{y}_t - \mathbf{H}_t\mathbf{x}_t)^\top.
\end{aligned} \tag{14}$$

The $\mathbf{w}_{1:T}$-subproblem is of the form

$$\min_{\mathbf{w}_{1:T}} \lambda \sum_{t=1}^{T} \|\mathbf{w}_t\|_1 + \frac{\rho}{2}\sum_{t=1}^{T} \|\boldsymbol{\Omega}_t\mathbf{x}_t - \mathbf{w}_t + \boldsymbol{\eta}_t\|^2, \tag{15}$$

which is non-differentiable. We update it by using a soft thresholding operation, which results from the computation of the appropriate proximal operator [18], given by

$$\mathbf{w}_t^{(k+1)} = \operatorname{sgn}(\mathbf{e}_t^{(k)}) \circ \max(|\mathbf{e}_t^{(k)}| - \lambda/\rho, 0), \tag{16}$$

where $\mathbf{e}_t^{(k)} = \boldsymbol{\Omega}_t\mathbf{x}_t^{(k+1)} + \boldsymbol{\eta}_t^{(k)}$, and sgn represents the signum function, and $\circ$ is the pointwise product. The resulting augmented Lagrangian Kalman smoother (ALKS) method is summarized in Algorithm 1.

Algorithm 1: Augmented Lagrangian Kalman smoother (ALKS)

```
Input: $\mathbf{y}_t$, $\mathbf{H}_t$, $\mathbf{A}_t$, $\mathbf{\Omega}_t$ for $t = 1, \ldots, T$; $\mathbf{m}_1$ and $\mathbf{P}_1$;
        parameters $\lambda$ and $\rho$.
Output: $\mathbf{x}^\star_{1:T}$, $\mathbf{Q}^\star$, and $\mathbf{R}^\star$.
Initialize: $\mathbf{x}^{(0)}_{1:T}$, $\mathbf{Q}^{(0)}$, $\mathbf{R}^{(0)}$, $\mathbf{w}^{(0)}_{1:T}$ and $\boldsymbol{\eta}^{(0)}_{1:T}$.
while not convergent do
    run the KS solver by (10) and (11);
    compute $\mathbf{x}_{1:T}$ by (12);
    learn $\mathbf{Q}$ and $\mathbf{R}$ by (14);
    compute $\mathbf{w}_{1:T}$ by (16);
    update $\boldsymbol{\eta}_{1:T}$ as in (7);
end
```

4. EXPERIMENTAL RESULTS

4.1. Simulated Estimation Scenario

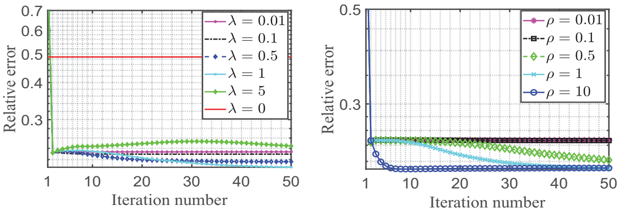
In this section, we present two sets of experiments to benchmark the performance of the proposed method. Firstly, we show the convergence curves of ALKS with different λ and ρ, and then compare ALKS with the existing variable splitting optimization methods such as the split Bregman method (SBM) [19], batch augmented Lagrangian method (b-ALM) [13], and first-order primal-dual algorithm (FOPD) [20].

To generate the ground-truth state $\mathbf{x}^{\text{true}}_{1:T}$, a random subset of nonzero coefficients is chosen. The measurement matrix $\mathbf{H}_t$ is randomly generated from the standard normal distribution, the state transition matrix $\mathbf{A}_t$ is sparse, $\mathbf{Q} = q\mathbf{I}$, and $\mathbf{R} = r\mathbf{I}$. We set $n_x = 30$, $n_y = 20$, $q = 0.1^2$, $r = 0.01^2$ and $T = 100$, and the number of non-zero elements in the ground truth is 5. The relative error is defined as

$$\sum_{t=1}^{T} \|\mathbf{x}^{(k)}_t - \mathbf{x}^{\text{true}}_t\|_2 \,/\, \sum_{t=1}^{T} \|\mathbf{x}^{\text{true}}_t\|_2 \qquad (17)$$

Using the parameters $\lambda = 0.01, 0.1, 0, 0.5, 1, 5$, we study the performance of ALKS in six different cases. Note that when $\lambda = 0$, the optimization problem (4) can be solved by KS. In Fig. 2, the vertical axis shows the estimating relative error and the horizontal axis shows the iteration number k. Not surprisingly, all six cases converge to the optimal values within tens of iterations, and a proper choice for λ has the influence on the relative error (see Fig. 2(a)). We now turn to study the parameter $\rho = 0.01, 0.1, 0.5, 1, 10$ in Fig. 2(b). The setting with $\rho = 10$ is the best for this case because of fastest convergence and lowest relative error. Then, we learn the covariance parameters q and r. In particular, the learning results of the parameter q are shown in Fig. 3. We observe that the value of q learned by different methods including SBM, b-ALM and ALKS are around 0.035, and the true $q = 0.01$.

Fig. 4 shows the relative error and the CPU time as functions of the iteration number. As can be seen, all the compared methods have roughly the same errors, and seem to converge with enough iterations. Running the SBM, b-ALM, and FOPD solvers is time-consuming. The b-ALM and ALKS solvers have the same convergence speed, but ALKS has a lower CPU time. We also increase the number of time steps T from 10^2 to 10^8. Table 1 reports the CPU times of the compared methods when the number of time steps T is varying. We used ten iterations for each method, which is in practice enough for convergence in these larger data sets. When $T = 10^5, 10^6, 10^7, 10^8$, the computer operations on SBM, b-ALM, and FOPD run out of memory, and do not report here. It can be seen that ALKS and KS significantly outperform the other variable splitting optimization methods with respect to the CPU time.



(a) (b)

Fig. 2: Performance as function of iteration number k in (a) different λ, with $\rho = 10$; and (b) different ρ, with $\lambda = 1$. The y-axis is in log-scale.

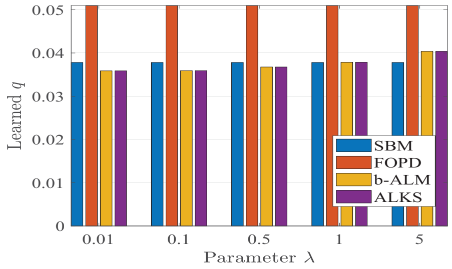
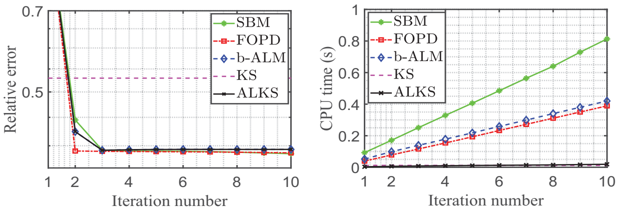


Fig. 3: Learned parameter q with respect to different λ.



(a) (b)

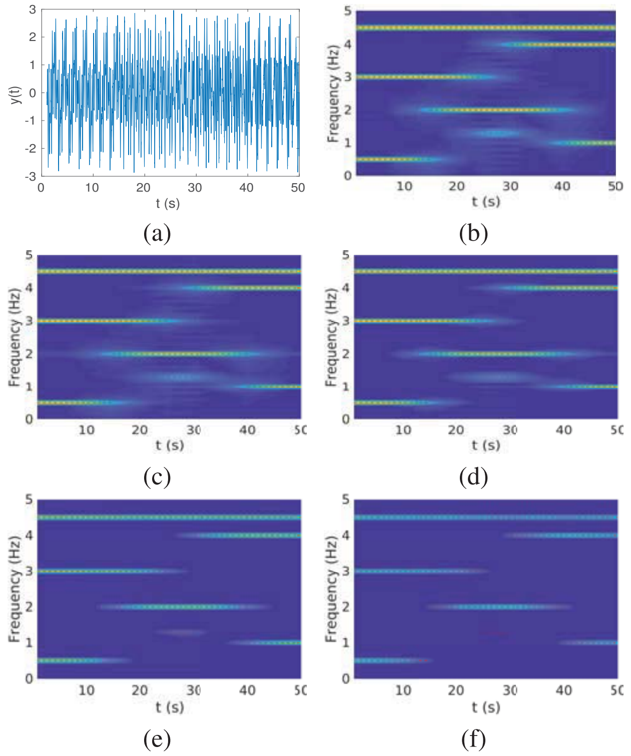
Fig. 4: Performance of different methods. (a) the relative error versus iteration number; (b) the average CPU time versus iteration number. The y-axis is in log-scale.

4.2. Sparse Spectro-temporal Estimation

The spectro-temporal (time-frequency) representation is a powerful tool of analyzing signals in time-varying frequency do-

Table 1: Average CPU time for different large-scale datasets in seconds.

T	10^2	10^3	10^4	10^5	10^6	10^7	10^8
SBM	0.58	421	86286	-	-	-	-
FOPD	0.54	199	42643	-	-	-	-
b-ALM	0.67	302	74286	-	-	-	-
KS	0.16	1.1	11.6	100	535	6520	21485
ALKS	0.19	1.56	13.3	134	767	12714	44971



(a) (b) (c) (d) (e) (f)

Fig. 5: Comparison of the spectro-temporal estimation of noisy sinusoidal data (a), by using the classical KS (b), ALKS with $\lambda = 0.01$ (c), ALKS with $\lambda = 0.2$ (d), ALKS with $\lambda = 1$ (e), ALKS with $\lambda = 0.01$ (f). The red dashed lines depict the ground truth frequency bands as defined in (19). The frequency range for estimation is set to be from 0.1 Hz to 5 Hz with resolution 0.1 Hz.

main. The classical estimation methods include short time Fourier transform (STFT) and continuous wavelet transform (CWT). In many practical applications such as biosignal [21] and audio processing [22], it is beneficial to obtain a sparse spectro-temporal representation of the signal. Inspired by recent studies [3, 21], the spectro-temporal estimation can be performed with KS. We can naturally combine our ALKS approach for such estimation with a sparse regularization. The idea in [21] is to model the time-varying signal $z(t)$ with a finite M-order Fourier series

$$z(t) = a_0 + \sum_{j=1}^{M} [a_j \cos(2\pi\, j\, f_0\, t) + b_j \sin(2\pi\, j\, f_0\, t)]\,, \quad (18)$$

where $\{a_j, b_j : j = 1, \cdots, M\}$ are the Fourier coefficients which we aim to estimate at time t. This can be reduced to estimation in a linear state-space model. The state vector $\mathbf{x}_t$ contains the Fourier coefficients at time t and $\mathbf{H}_t$ the basis functions. The other settings $\mathbf{A}$, $\mathbf{H}$, and $\mathbf{Q}(q)$, we refer the reader to [3] for details.

Let us first consider an ensemble of noisy sinusoidal data

$$\begin{aligned} y_t &= \varepsilon_t + \sin(2\pi\, 4.5\, t) \\ &+ \begin{cases} \sin(2\pi\, 0.5\, t) + \sin(2\pi\, 3\, t), & 1 \le t < 15 \\ \sin(2\pi\, 2\, t) + \sin(2\pi\, 3\, t), & 15 \le t < 25 \\ \sin(2\pi\, 1.3\, t) + \sin(2\pi\, 2\, t), & 25 \le t < 30 \\ \sin(2\pi\, 2\, t) + \sin(2\pi\, 4\, t), & 30 \le t < 40 \\ \sin(2\pi\, 1\, t) + \sin(2\pi\, 4\, t), & 40 \le t < 50 \end{cases}, \end{aligned} \quad (19)$$

where $\varepsilon_t \sim \mathcal{N}(0, 0.1^2)$, and we sample the data at a frequency of 100 Hz. In this case, the ideal spectro-temporal representation needs to be sparse, as the spectral features should be distributed in certain frequencies, for example, 0.5 Hz, 3 Hz, and 4.5 Hz as defined in (19).

We first perform spectro-temporal estimation using KS as proposed in [3]. In Fig. 5 (b), it shows that the classical KS indeed can represent the frequency features with a good resolution. However, we notice that there are noisy harmonics manifestation on the break/jump-points of those frequency lines, for example, when $10 < t < 40$ s.

Those harmonic noise problems can be alleviated by postulating an L_1-regularization on $\mathbf{x}_t$, that we demand a sparse spectro-temporal representation. The regularized estimation can be solved using the proposed ALKS solver. The estimation results are shown in the Figs. 5 (c), (d), (e), and (f), where we control the global weight parameter $\lambda = 0.01$, 0.2, 1, and 2, respectively. When $\lambda = 0.01$, the result is almost the same to the one without regularization (Fig. 5 (b)). If we increase the parameter λ to 0.2 and 1, we find that the unnecessary harmonics noise are reduced. The setting $\lambda = 2$ leads to over-regularization, as the frequency feature at 1.3 Hz disappears.

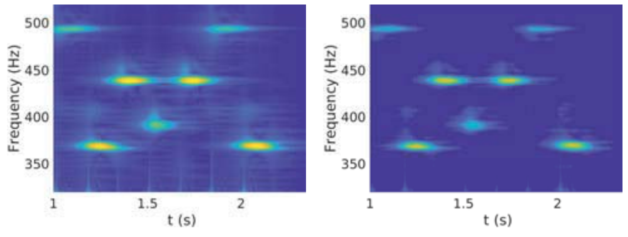
The estimation results for the parameters r and q are shown in Table 2. We observe that the value of q decreases as λ increases, in contrast to the evolution of r. This phenomena is expected, because the L_1 term acts stronger with larger weight λ. It is a trade-off between the sparsity and details level of the estimation. When $\lambda = 0.01$ is small, the estimated $r = 0.02$ is very close to the true noise setting $\varepsilon_t \sim \mathcal{N}(0, 0.1^2)$ in (19).

Table 2: Estimated parameters r and q with respect to λ.

λ	0.01	0.1	0.2	0.5	1	2
q	3.274	3.133	2.976	2.589	2.051	1.189
r	0.020	0.039	0.066	0.144	0.251	0.516

4.3. Application to Automatic Music Transcription

Sparse spectro-temporal features are especially useful in music production. The aim is to automatically transcribe (recognise) the music notes from audio signal. This procedure can be done with machine learning algorithms and time-frequency features [23], and a less noisy representation could be expected to improve the performance of the identification procedure. To demonstrate the capability of ALKS for real audio data, we take the first seven notes from the Lute Suite in E minor, BWV 996 by J. S. Bach, and generate an audio signal using a synthesizer.



(a) (b)

Fig. 6: The musical note recognition using spectro-temporal estimation. The seven notes are B4, F4$^\sharp$, A4, G4, A4, B4, and F4$^\sharp$ with parameters $\lambda = 0.01$, $\rho = 100$, and 10 epochs by using (a) classical KS, (b) the ALKS solver.

The spectro-temporal estimation on the audio using both classical KS and our ALKS solver are shown in the Fig. 6. We find that the frequency features of those notes are well estimated using both of the two methods. However, the result of KS is contaminated by harmonic frequency noises. This might be problematic if one needs to utilise the machine learning algorithms to recognise the notes from those spectral images [23]. As shown in Fig. 6 (b), using ALKS solver, the unnecessary harmonic noises are reduced, while the main features of notes are preserved.

5. CONCLUSION

In this article, we have proposed an augmented Lagrangian Kalman smoother method for solving the state estimation and parameter learning problem under a sparse state assumption. For the general L_1-penalized optimization problem, we propose a combination method of KS and the augmented Lagrangian method, where KS is used to solve the primal variable update subproblems. Experiments demonstrate that the proposed method has a lower computational complexity than the state-of-the-art methods. The performance has been illustrated in practical sparse spectro-temporal estimation.

6. REFERENCES

[1] Y. B. Shalom, X. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, Wiley, 2001.

[2] S. Särkkä, *Bayesian Filtering and Smoothing*, Cambridge, U.K.: Cambridge Univ. Press, Aug. 2013.

[3] Z. Zhao, S. Särkkä, and A. B. Rad, “Kalman-based spectro-temporal ECG analysis using deep convolutional networks for atrial fibrillation detection,” *arXiv preprint arXiv:1812.05555*, 2018.

[4] N. Vaswani, “Kalman filtered compressed sensing,” *Proc. IEEE Int. Conf. Image Processing (ICIP)*, pp. 893–896, Oct. 2008.

[5] A. S. Charles, A. Balavoine, and C. J. Rozell, “Dynamic filtering of time-varying sparse signals via L1 minimization,” *IEEE Trans. Signal Process.*, vol. 64, no. 21, pp. 5644–5656, Nov. 2016.

[6] A. Aravkin, J. V. Burke, L. Ljung, A. Lozano, and G. Pillonetto, “Generalized Kalman smoothing: Modeling and algorithms,” *Automatica*, vol. 86, pp. 63–86, Dec. 2017.

[7] M. Elad, P. Milanfar, and R. Rubinstein, “Analysis versus synthesis in signal priors,” *Inv. Probl.*, vol. 23, no. 3, pp. 947–968, Sept. 2007.

[8] R. Gao, S. A. Vorobyov, and H. Zhao, “Image fusion with cosparse analysis operator,” *IEEE Signal Process. Lett.*, vol. 24, no. 7, pp. 943–947, July 2017.

[9] R. Gao, F. Tronarp, and S. Särkkä, “Iterated extended kalman smoother-based variable splitting for L1-regularized state estimation,” *to appear in IEEE Trans. Signal Process.*, 2019.

[10] H. Cox, “On the estimation of state variables and parameters for noisy dynamic systems,” *IEEE Trans. Automat. Control*, vol. 9, no. 1, pp. 5–12, 1964.

[11] R. Glowinski, S. J. Osher, and W. Yin, *Splitting Methods in Communication, Imaging, Science, and Engineering.*, Cham, Switzerland: Springer, 2017.

[12] N. K. Dhingra, S. Z. Khong, and M. R. Jovanović, “A second order primal-dual algorithm for nonsmooth convex composite optimization,” *IEEE 56th Annual Conf. on Decision and Control (CDC)*, pp. 2868–2873, Dec. 2017.

[13] C. Wu and X. C. Tai, “Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models,” *SIAM J. Imaging Sci.*, vol. 3, no. 3, pp. 300–339, 2010.

[14] C. Wu, J. Zhang, and X.-C. Tai, “Augmented Lagrangian method for total variation restoration with non-quadratic fidelity,” *Inv. Probl.*, vol. 5, pp. 237–261, 2011.

[15] R. Gao, F. Tronarp, and S. Särkkä, “Combined analysis-L1 and total variation ADMM with applications to MEG brain imaging and signal reconstruction,” in *26th European Signal Proces. Conf. (EUSIPCO)*, Roma, Italy, Sept. 2018, pp. 1930–1934, IEEE.

[16] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine Learning.*, vol. 3, no. 1, pp. 1–122, 2011.

[17] H. E. Rauch, F. Tung, and C. T. Striebel, “Maximum likelihood estimates of linear dynamic system,” *AIAA J.*, vol. 3, no. 8, pp. 1445–1450, Aug. 1965.

[18] N. Parikh and S. Boyd, “Proximal algorithms,” *Found. Trends Optim.*, vol. 1, no. 3, pp. 123–231, 2013.

[19] T. Goldstein and S. Osher, “The split Bregman method for L1-regularized problems,” *SIAM J. Imaging Sci.*, vol. 2, no. 2, pp. 323–343, Apr. 2009.

[20] A. Chambolle and T. Pock, “A first-order primal-dual algorithm for convex problems with applications to imaging,” *J. Math. Imaging. Vis.*, vol. 40, no. 1, pp. 120–145, May 2011.

[21] Z. Zhao, S. Särkkä, and A. B. Rad, “Spectro-temporal ECG analysis for atrial fibrillation detection,” in *2018 IEEE 28th International Workshop on Mach. Learn. for Signal Process. (MLSP)*, Sept. 2018, pp. 1–6.

[22] T. J. Gardner and M. O. Magnasco, “Sparse time–frequency representations,” *Proceedings of the National Academy of Sciences*, vol. 103, no. 16, pp. 6094–6099, 2006.

[23] S. Sigtia, E. Benetos, and S. Dixon, “An end-to-end neural network for polyphonic piano music transcription,” *IEEE/ACM Trans. Audio Speech and Language Process.*, vol. 24, no. 5, pp. 927–939, May 2016.