State Smoothing by Sum-of-Norms Regularization

Henrik Ohlsson, Fredrik Gustafsson, Lennart Ljung and Stephen Boyd

***Abstract*— The presence of abrupt changes, such as impulsive disturbances and load disturbances, make state estimation considerably more difficult than the standard setting with Gaussian process noise. Nevertheless, this type of disturbances is commonly occurring in applications which makes it an important problem. An abrupt change often introduces a jump in the state and the problem is therefore readily treated by change detection techniques. In this paper, we take a rather different approach. The state smoothing problem for linear state space models is here formulated as a least-squares problem with sum-of-norms regularization, a generalization of the ℓ_1-regularization. A nice property of the suggested formulation is that it only has one tuning parameter, the regularization constant which is used to trade off fit and the number of jumps.**

I. Introduction

We consider the problem of state estimation in linear state space models, where impulsive disturbances occur in the process model. There are several conceptually different ways to handle disturbances in state estimation. One possibility is to model the disturbance as a sequence of Gaussian random variables with known second order moment, so the optimal solution is provided by the Kalman filter (KF), see [17], [16]. Another, quite different, possibility is to assume that the disturbance is a deterministic arbitrary sequence, and apply subspace projections. See [15] for details and more references on that approch. The case of impulsive process noise occurs frequently in at least three different application areas.

- In automatic control, impulsive noise is often used to model load disturbances.
- In target tracking, impulsive noise is used to model force disturbances, corresponding to maneuvers for the tracked object.
- In fault detection and isolation (FDI) literature impulsive noise is used to model additive faults. Usually, this is done in a deterministic setting [25], but a stochastic framework is also common [2], [12].

We formulate the problem in a probabilistic framework where the KF is the best linear unbiased estimator (BLUE), and the interacting multiple model (IMM, [4]) algorithm provides an approximation to the exact problem. In contrast to IMM, we here propose a method that solves an approximate problem in an optimal way.

Partially supported by the Swedish foundation for strategic research in the center MOVIII and by the Swedish Research Council in the Linnaeus center CADICS.

H. Ohlsson, F. Gustafsson and L. Ljung are with the Division of Automatic Control, Department of Electrical Engineering, Linköping University, Sweden, {ohlsson,fredrik,ljung}@isy.liu.se.

Stephen Boyd is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA, boyd@stanford.edu.

Our approach is based on convex optimization. It is well-known that the KF solves an optimization problem where the sum of squared two-norms of the process and measurement noises is minimized. Inspired by the recent progress of sum-of-norms regularization in the statistical literature ([18], see also related contribution in the control community [23]), we suggest to change the squared two-norm of the process noise to a sum-of-norms, to capture the impulse character of the process disturbance. The consequence of this is that a sparse sequence of process noise is automatically obtained in contrast to the KF. The algorithm solves the smoothing problem in linear complexity, and a further advantage compared to KF and IMM algorithms is that convex constraints of the state sequence are easily handled in the same framework.

We start with a brief introduction to dynamical systems and stochastic disturbances. This is followed up by a discussion on the smoothing problem in Section III. In particular, we care about the optimization formulation of the Kalman smoother. Section IV contains the main contribution of the paper, the proposed method for state smoothing with impulsive process disturbances. We call the method state smoothing by sum-of-norms regularization (STATESON). In Section V a comparison with popular methods for state smoothing with impulsive process noise is given. A justification by some numerical illustrations is given in Section VI. The paper is ended by a conclusion in Section VII.

II. Introduction: Dynamic Systems with Stochastic Disturbances

The standard linear state space model with stochastic disturbances is well known to be

$$\begin{aligned} x(t+1) &= A_t x(t) + B_t u(t) + G_t v(t) \\ y(t) &= C_t x(t) + e(t). \end{aligned} \tag{1a}$$

Here, v and e are white noises: sequences of independent random vectors

$$\begin{aligned} E[v(t)] = 0, &\quad E[e(t)] = 0 \quad \forall t \\ E[v(t)v^T(s)] = 0, &\quad E[e(t)e^T(s)] = 0 \quad \text{if } t \neq s \\ E[v(t)v^T(t)] = R_1(t), &\quad E[e(t)e^T(t)] = R_2(t). \end{aligned} \tag{1b}$$

The independence of the noise sequences is required in order to make x a *state* or a Markov process.

The model (1) with the "process noise" v being Gaussian is a standard model for control applications. v then represents the combined effect of all those non-measurable inputs that in addition to u affect the states. This is the common model used both for state estimation and control design based on LQG.

But, an equally common situation is that v corresponds to an *unknown input*. It could be

- a *load disturbance* e.g. a step change in moment load in an electric motor, a (up or down) hill for a vehicle, etc. (Sometimes, the term load disturbance is used only for the case $B_t = G_t$.)
- an event that causes the state to jump, *a change*, see e.g. [12].

Such unknown inputs are not naturally modeled as Gaussian noise. Instead it is convenient to capture their unpredictable nature by the distribution (cf eq (2.10)-(2.11) in [20].)

$$v(t) = \delta(t)\eta(t) \tag{2a}$$

where

$$\delta(t) = \begin{cases} 0 & \text{with probability } 1-\mu \\ 1 & \text{with probability } \mu \end{cases} \tag{2b}$$

$$\eta(t) \sim N(0, Q) \tag{2c}$$

This makes $R_1(t) = \mu Q$. The matrices A_t and G_t may further model the waveform of the disturbance as a response to the pulse in v.

We shall in this contribution discuss efficient ways of estimating the states under this assumption of the process noise v.

III. State Estimation (Smoothing)

A natural and common problem is to estimate the states $x(t)$ of the system (1) from measurements of u and y. For the case of Gaussian noise sources, this problem is of course solved by the Kalman filter and the Kalman smoother, [16]. For our current purpose it is of interest to view this as an explicit minimization problem. For given $x(1)$ the states $x(t), t = 2, \dots, N$ can be computed from $v(t), t = 1, \dots, N-1$. The quality of these states could be measured by the criterion of fit to observations:

$$\sum_{t=1}^{N} \|R_2^{-1/2}(t)(y(t) - C_t x(t))\|_2^2 \tag{3}$$

where, for a vector $z = [z_1\, z_2\, \dots\, z_{n_z}]^T$, $\|z\|_p \triangleq (\sum_{i=1}^{n_z} |z_i|^p)^{1/p}$. This should be minimized w.r.t. $x(1), v(t);\ t = 1, \dots, N-1$. At the same time, the use of jumps in the states, corresponding to v should be constrained in some sense, so as to avoid over-fitting to the noisy y-measurements.

The most common way is to use a quadratic regularization

$$\begin{aligned} \min_{x(1),v(t),1\le t\le N-1} & \sum_{t=1}^{N} \|R_2^{-1/2}(t)(y(t) - C_t x(t))\|_2^2 \\ & + \sum_{t=1}^{N-1} \|R_1^{-1/2}(t) v(t)\|_2^2 \end{aligned} \tag{4}$$

which gives the classical Kalman smoothing estimate, e.g. [16]. In the case of Gaussian process noise v, this is also the Maximum likelihood estimate and gives the conditional mean of $x(t)$ given the observations. It is a pure least squares problem, and the solution is usually given in various recursive filter forms, see e.g. [21].

Since $x(t)$ is a given function of $x(1), v(t)$ and the known sequence $u(t)$, it may seem natural to do the minimization directly over $x(t)$, i.e. to write

$$\begin{aligned} \min_{x(t),1\le t\le N} & \sum_{t=1}^{N} \|R_2^{-1/2}(t)(y(t) - C_t x(t))\|_2^2 \\ & + \sum_{t=1}^{N-1} \|R_1^{-1/2}(t) G_t^{\dagger}(x(t+1) - A_t x(t) - B_t u(t))\|_2^2 \end{aligned} \tag{5a}$$

where $G^{\dagger}$ is the pseudo inverse of G. However, if G is not full rank, nothing constrains the state evolution in the null space of G, so (5a) must be complemented with the constraint

$$G_t^{\perp}(x(t+1) - A_t x(t) - B_t u(t)) = 0 \tag{5b}$$

where $G^{\perp}$ is the projection onto the null-space of G,

$$G^{\perp} \triangleq I - GG^{\dagger}. \tag{6}$$

However, since several approaches can be interpreted as explicit methods to estimate $v(t)$ (or $\delta(t)$ in (2)), we shall adhere to the (equivalent) formulation (4).

IV. The Proposed Method: State Smoothing by Sum-of-Norms Regularization

The type of process noise that we are interested in (see (2)) motivates a rather different regularization term than the one used in (5a).

A. Sum-of-Norms Regularization

To penalize state changes over time, we use a penalty or regularization term (see e.g. [6, p.308]) that is a sum of norms of the estimated extra inputs $v(t)$:

$$\begin{aligned} \min_{x(1),v(t),1\le t\le N-1} & \sum_{t=1}^{N} \|R_2^{-1/2}(t)(y(t) - C_t x(t))\|_2^2 \\ & + \lambda \sum_{t=1}^{N-1} \|Q^{-1/2}(t) v(t)\|_p \end{aligned} \tag{7a}$$

subject to

$$x(t+1) = A_t x(t) + B_t u(t) + G_t v(t) \tag{7b}$$

where the ℓ_p-norm is used for regularization, and λ is a positive constant that is used to control the trade-off between the fit to the observations $y(t)$ (the first term) and the size of the state changes caused by $v(t)$ (the second term). The regularization norm could be any ℓ_p-norm, like ℓ_1 or ℓ_2, but it is crucial that it is a sum of norms, and not a sum of squared norms, which was used in (5a).

When the regularization norm is taken to be the ℓ_1 norm, i.e., $\|z\|_1 = \sum_{i=1}^{n_z} |z_i|$, the regularization in (7a) is a standard ℓ_1 regularization of the least-squares criterion. Such regularization has been very popular recently, e.g. in the much used Lasso method, [29] or compressed sensing [9], [7].

There are two key reasons why the criterion (7a) is attractive:

- It is a convex optimization problem, so the global solution can be computed efficiently. In fact, its special structure allows it to be solved in $O(N)$ operations, so it is quite practical to solve it for a range of values of λ, even for large values of N.
- The sum-of-norms form of the regularization favors solutions where "many" (depending on λ) of the regularized variables come out as exactly zero in the solution. In this case, this implies that many of the estimates of $v(t)$ become zero (with the number of $v(t)$:s becoming zero controlled roughly by λ).

A third advantage is that

- It is easy to include realistic state constraints without destroying convexity.

We should comment on the difference between using an ℓ_1 regularization and some other type of sum-of-norms regularization, such as sum-of-Euclidean norms. With ℓ_1 regularization, we obtain an estimate of v having many of its components equal to zero. When we use sum-of-norms regularization, the whole estimated vector $v(t)$ often becomes zero; but when it is nonzero, typically all its components are nonzero. In a statistical linear regression framework, sum-of-norms regularization is called Group-Lasso [30], since it results in estimates in which many groups of variables are zero.

B. Regularization Path and Critical Parameter Value

The estimated sequence $v(t)$ as a function of the regularization parameter λ is called the *regularization path* for the problem. Roughly, larger values of λ correspond to estimated $x(t)$ with worse fit, but an estimate of $v(t)$ with many zero elements. A basic result from convex analysis tells us that there is a value $\lambda^{\max}$ for which the estimated $v(t)$ is identically zero if and only if $\lambda \geq \lambda^{\max}$. In other words, $\lambda^{\max}$ gives the threshold above which $v(t) = 0$, $t = 1, ..., N$. The critical parameter value $\lambda^{\max}$ is very useful in practice, since it gives a very good starting point in finding a suitable value of λ. Reasonable values are typically in the order of $0.01\lambda^{\max}$ to $\lambda^{\max}$.

Proposition 1: Introduce ε_t for the (process) noise free residual

$$\varepsilon_t \triangleq y(t) - C_t\Big(\sum_{r=1}^{t-1}\prod_{s=r+1}^{t-1} A_s B_r u(r) + \left[\prod_{s=1}^{t-1} A_s\right] x_1\Big) \quad (8)$$

and take x_1 to minimize

$$\sum_{t=1}^{N} \left\| R_2^{-1/2}(t)\varepsilon_t \right\|_2^2. \quad (9)$$

We can then express $\lambda^{\max}$ as

$$\lambda^{\max} = \max_{k=1,\dots,N-1} \left\| 2Q^{1/2} \sum_{t=k+1}^{N} \varepsilon_t R_2^{-1}(t)(C_t \prod_{s=k+1}^{t-1} A_s G_k)^T \right\|_q \quad (10)$$

with $\|\cdot\|_q$ the dual norm ($\frac{1}{p}+\frac{1}{q}=1$) associated with $\|\cdot\|_p$ used in (7a).

The proof is given in the appendix.

C. Iterative Refinement

To (possibly) get even more zeros in the estimate of $v(t)$, with no or small increase in the fitting term, iterative reweighting can be used [8]. We modify the regularization term in (7a) and consider

$$\begin{aligned} \min_{x(1),v(t),1\leq t\leq N-1} & \sum_{t=1}^{N} \|R_2^{-1/2}(t)(y(t) - C_t x(t))\|_2^2 \\ & + \lambda \sum_{t=1}^{N-1} \alpha(t) \|Q^{-1/2} v(t)\|_p \end{aligned} \quad (11)$$

where $\alpha(1), \dots, \alpha(N-1)$ are positive weights used to vary the regularization over time. Iterative refinement proceeds as follows. We start with all weights equal to one i.e., $\alpha^{(0)}(t) = 1$. Then for $i = 0, 1, \dots$ we carry out the following iteration until convergence (which is typically in just a few steps).

1) *Find the state estimate.*
   Compute the optimal $v^{(i)}(t)$ using (11) with the weighted regularization using weights $\alpha^{(i)}$.
2) *Update the weights.*
   For $t = 1, \dots, N-1$, set $\alpha^{(i+1)}(t) = 1/(\epsilon + \|Q^{-1/2} v^{(i)}\|_p)$.

Here ϵ is a positive parameter that sets the maximum weight that can occur.

One final step is also useful. From our final estimate of $v(t)$, we simply define set of times T at which an estimated load disturbance occurs (i.e., $T = \{t | v(t) \neq 0\}$) and carry out a final optimization over just $v(t), t \in T$.

Remark 1: If the jump covariance Q in (2) is known or can be given a good value, the final optimization step should be replaced by a Kalman smoother with the time-varying process noise

$$R_1(t) = \begin{cases} 0 & \text{for } t\text{:s such that } v(t) = 0 \\ Q & \text{otherwise.} \end{cases}$$

It should be noticed that if the correct jump-times and Q have been found, this is actually optimal in the sense that no other smoother (linear or nonlinear) can achieve an unbiased estimate with a lower error covariance.

D. Solution Algorithms and Software

Many standard methods of convex optimization can be used to solve the problem (7a). Systems such as CVX [11], [10] or YALMIP [22] can readily handle the sum-of-norms regularization, by converting the problem to a cone problem and calling a standard interior-point method. For the special case when the ℓ_1 norm is used as the regularization norm, more efficient special purpose algorithms and software can be used, such as `l1_ls` [19]. Recently many authors have developed fast first order methods for solving ℓ_1 regularized problems, and these methods can be extended to handle the sum-of-norms regularization used here; see, for example, [27, §2.2]. Both interior-point and first-order methods have a complexity that scales linearly with N.

V. Other Approaches

Many methods for state estimation with non-Gaussian noise as in (2) are suggested in the literature, both in connection with change detection, e.g. [12], and target tracking, e.g. [1]. Many of them can be seen as ways to explicitly estimate $v(t)$ or $\delta(t)$ in (2).

If the δ-sequence was known, the problem could be treated as a Kalman smoother with known, time varying $R_1(t)$ ($R_1(t_k) = Q$ for those t_k with $\delta(t_k) = 1$ and $R_1(t_\ell) = 0$ otherwise.) This is sometimes known as the *Clairvoyant filter* (or filter with an oracle). The (time-varying) smoothed state error covariance matrix can readily be computed for this case. Clearly this gives a lower bound for any possible estimate, which no other (linear or nonlinear) filter can beat.

Based on the model (2), a number of approximative methods have been developed. If $\delta(t)$ is not known, we could hypothesize in each time step that it is either 0 or 1. This leads to a large bank (2^N) of Kalman filters as the optimal solution. The posterior probability of each filter can be estimated from this bank, which consists of a weighted sum of the state estimates from each filter. In practice, the number of filters in the bank must be limited, and there are two main options (see Chapter 10 in [13]):

- Merging trajectories of different $\delta(t)$ sequences. This includes the well-known IMM filter, see [4].
- Pruning, where unlikely sequences are deleted from the filter bank.

Change detection techniques can also be used to detect the time instances when $\delta(t) = 1$. In the linear case, e.g., a change detection algorithm can be applied to the innovations of a Kalman filter to detect jumps in the process noise. If a jump is detected, the process noise covariance in the Kalman filter is made e.g. 10 times larger to adapt to the abrupt state change.

VI. Numerical Illustration

Consider the discrete time model of a DC motor (see e.g. [14], $T_s = 0.1$s, angle velocity and angle as state)

$$\begin{aligned} x(t+1) &= \begin{bmatrix} 0.7047 & 0 \\ 0.08437 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 11.81 \\ 0.625 \end{bmatrix} (u(t) + v(t)) \\ s(t) &= \begin{bmatrix} 0 & 1 \end{bmatrix} x(t) \\ y(t) &= s(t) + e(t), \end{aligned} \tag{12}$$

with $u(t)$ a ± 0.1 PRBS signal (Pseudorandom binary sequence), $e(t) \sim N(0, 1)$ and $x(1) \sim N(0, I)$. The system was simulated with $v(t)$ distributed according to (2) with $\mu = 0.015$ and $Q = 0.5$, which gave the particular load disturbance sequence

$$v(t) = \begin{cases} -0.6 & \text{for } t = 55, \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

and $y(t), t = 1, \ldots, 100$, was observed. The resulting estimate of the angle $s(t)$ using $\lambda = 0.1\lambda^{\max}$, two refinement iterations, $Q = 0.5$, $R_2 = 1$ and an Euclidean norm in the regularization is shown in Figure 1. Figure 1 also shows the measurement $y(t)$ and the true sequence $s(t)$ that was used to generate the $y(t)$ measurements. $v(t)$ was estimated to

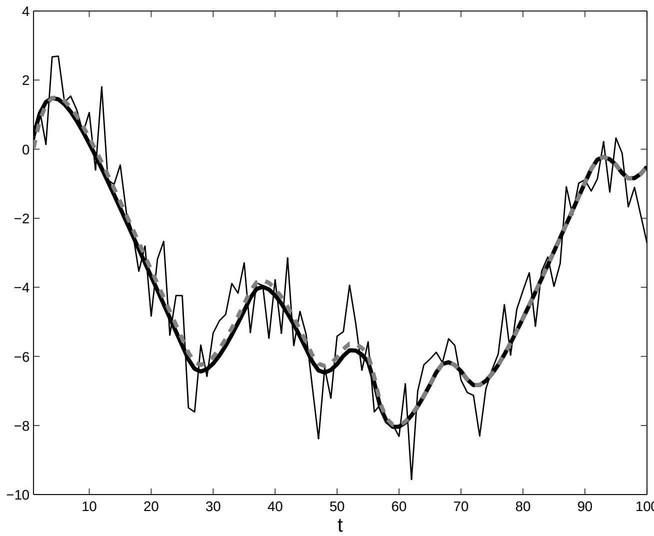

Fig. 1. The resulting estimate of $s(t)$ showed with a solid thick line; dashed line, true sequence $s(t)$; solid thin black line, measurements $y(t)$. The jump in $v(t)$ at $t = 55$ is hardly visible.

$$v(t) = \begin{cases} -0.55 & \text{for } t = 55, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

The mean squared error (MSE) for the state estimate was for this particular setup 0.28. Since the jump-time was correctly found, the estimate almost coincide with the estimate of the Clairvoyant estimator. If a Kalman smoother is applied with the true measurement and process noise variances ($R_1 = \mu Q = 0.0075$, $R_2 = 1$) a MSE of 1.0 was obtained. The result is summarized in Table I.

TABLE I

MSE for the state estimate obtained by the Kalman smoother (BLUE), state estimation by sum-of-norms regularization (STATESON) and Clairvoyant smoother.

Algorithm	MSE
BLUE	1.0
STATESON	0.28
Clairvoyant smoother	0.12

Let us now compare the state smoothing by sum-of-norms regularization (7) with some other methods that have been suggested in the literature.

For the same sequence $v(t)$ we run Monte-Carlo simulations over realizations of the measurement noise $e(t)$ with $R_2 = 1$. We run 2000 simulations and compute the smoothed state squared error over the runs.

We compare:

1) The proposed method state smoothing by sum-of-norms regularization.
2) Conventional Kalman smoother with $R_1 = 0.0075$ and $R_2 = 1$.

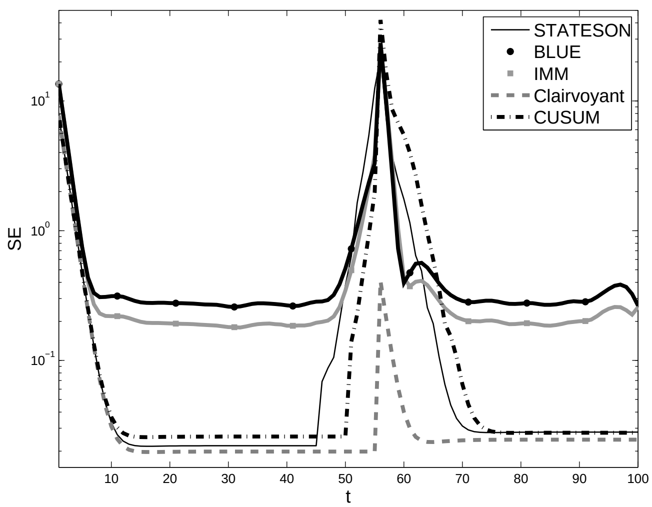

Fig. 2. Mean (over Monte Carlo runs) squared errors (SE) versus time. All the sample SE means were taken over Monte Carlo runs and not time. 2000 Monte Carlo runs were used.

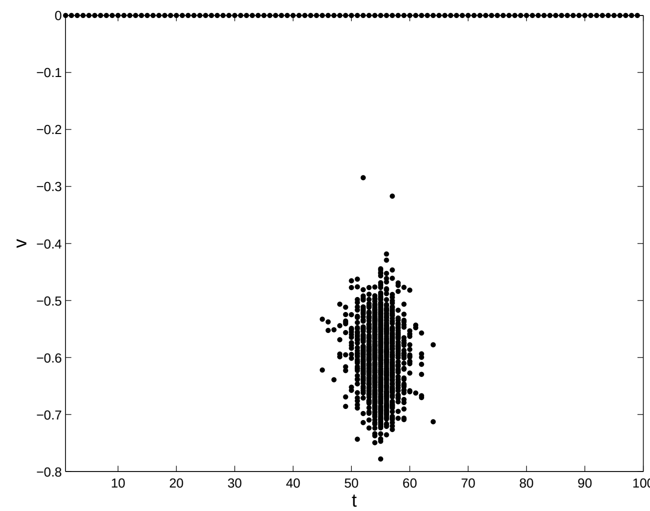

Fig. 3. The STATESON estimates of v that gave the squared errors in Figure 2.

3) Kalman smoother together with CUSUM (Cumulative Sum [24]). First, CUSUM was applied to both the whitened innovations and the negative whitened innovations of a Kalman filter with $R_2 = 1$. R_1 was set to 0.5 when the test statistic exceeded a threshold h but was otherwise 0. In a second step, a Kalman smoother was applied with $R_1 = 0.5$ at the time instances of detected changes and $R_1 = 0$ otherwise. $h = 10$ and drift term $\gamma = 1$ gave good performance.
4) IMM smoother with two modes, $R_2 = 1$ for both and $R_1 = 0$ and 0.5, respectively, with probabilities 0.985 and 0.015. The IMM smoothing implementation of [28] was used.
5) The lower bound according to the Clairvoyant smoother.

See Figure 2. We see that state smoothing by sum-of-norms regularization outperforms the Kalman smoother and the IMM smoother and that it gets fairly close to the lower bound given by the Clairvoyant smoother. Kalman smoother together with CUSUM does almost as good as state smoothing by sum-of-norms regularization. Figure 3 shows a plot of estimated v values by STATESON for the 2000 runs.

Let us also investigate how the methods perform under varying signal-to-noise conditions (the design parameters held fixed over the different SNRs, tuned for a SNR of 5.7). 200 Monte Carlo runs were performed for a number of different R_2 to produce the plot shown in Figure 4. The mean of the squared errors were take both over time and the 200 Monte Carlo runs with the same signal-to-noise ration (SNR). The SNR was computed as

$$SNR = \left(\frac{\sum_{t=1}^{N} |s(t)|}{\sum_{t=1}^{N} |e(t)|} \right)^2 \tag{15}$$

where $s(t)$ is the signal given in (12) if $u(t) \equiv 0$ is feeding (12) and $e(t)$ is the measurement noise. The plot shows that

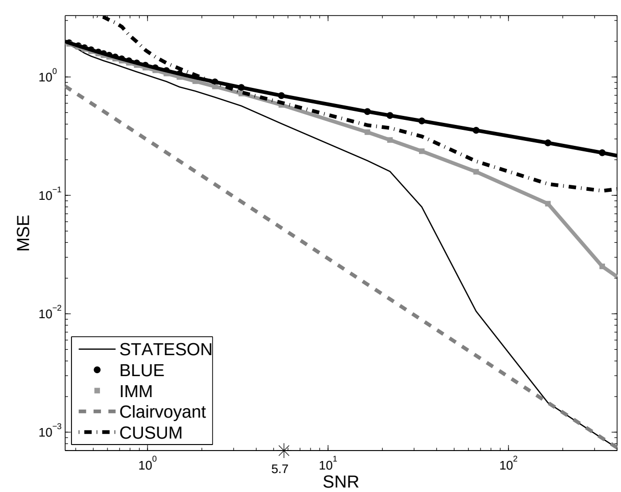

Fig. 4. MSE versus SNR. All the means were taken over Monte Carlo runs and time. 200 Monte Carlo runs were used for each SNR value. The simulations shown in Figure 2 had a SNR of 5.7 (in average).

state smoothing by sum-of-norms regularization does well in comparison with the compared methods. It may seem surprising that the Kalman smoother together with CUSUM dose not do better. CUSUM detects the changes in most cases, however, it does not give an accurate estimate of the time of the changes (not even in high SNR). CUSUM also suffers of varying SNR and would do better if it was retuned for each new SNR value.

VII. Conclusion

A novel formulation of the state estimation problem in the presence of abrupt changes has been presented. The proposed approach treats the state smoothing problem as a constrained least-squares problem with a sum-of-norms regularization. Some numerical illustrations have been given.

The approach can be seen an extension of the technique used for segmentation of ARX-models in [23]. The extension to nonlinear models is seen as a potential future research direction.

APPENDIX

To verify our formula for $\lambda^{\max}$ we use convex analysis [26], [3], [5]. Fist note that

$$\begin{aligned} x(t) =& G_{t-1}v(t-1) + A_{t-1}x(t-1) + B_{t-1}u(t-1) \\ =& \sum_{r=1}^{t-1}\left(\prod_{s=r+1}^{t-1} A_s\right)(G_r v(r) + B_r u(r)) + \left(\prod_{s=1}^{t-1} A_s\right) x_1. \end{aligned}$$

The objective in (7a) can then be written as

$$\begin{aligned} \sum_{t=1}^{N} \left\| R_2^{-1/2}(t) \left(\varepsilon_t - C_t \sum_{r=1}^{t-1} \left(\prod_{s=r+1}^{t-1} A_s \right) G_r v(r) \right) \right\|^2 \\ + \lambda \sum_{t=1}^{N-1} \|Q^{-1/2} v(t)\|_p. \end{aligned} \tag{16}$$

using (8). Since the gradient of $\|Q^{-1/2}v(t)\|_p$ w.r.t. $Q^{1/2}v(t)$ must lie in the unit ball in the dual norm, i.e.

$$\left\| Q^{1/2} \nabla_{v(t)} \|Q^{-1/2} v(t)\|_p \right\|_q \leq 1, \; t = 1, \dots, N-1 \tag{17}$$

it is enough that λ exceeds the q-norm of

$$Q^{1/2} \nabla_{v(k)} \sum_{t=1}^{N} \left\| R_2^{-1/2}(t) \left(\varepsilon_t - C_t \sum_{r=1}^{t-1} \left(\prod_{s=r+1}^{t-1} A_s \right) G_r v(r) \right) \right\|^2 \tag{18}$$

evaluated at $v(t) = 0, t = 1, \dots, N-1$ and for all k to guarantee that $v(t) = G_t^{\dagger}(x(t+1) - A_t x(t) - B_t u(t)) = 0, t = 1, \dots, N-1$ minimizes (7a). The gradient can be computed to

$$-2 \sum_{t=k+1}^{N} \left(\varepsilon_t - C_t \sum_{r=1}^{t-1} \left(\prod_{s=r+1}^{t-1} A_s \right) G_r v(r) \right) R_2^{-1}(t) (C_t \prod_{s=k+1}^{t-1} A_s G_k)^T \tag{19}$$

and evaluated at $v(t) = 0, t = 1, \dots, N-1$ we obtain

$$-2 \sum_{t=k+1}^{N} \varepsilon_t R_2^{-1}(t) (C_t \prod_{s=k+1}^{t-1} A_s G_k)^T. \tag{20}$$

Hence, as stated in Proposition 1, $\lambda^{\max}$ is given by

$$\lambda^{\max} = \max_{k=1,\dots,N-1} \left\| 2Q^{1/2} \sum_{t=k+1}^{N} \varepsilon_t R_2^{-1}(t) (C_t \prod_{s=k+1}^{t-1} A_s G_k)^T \right\|_q. \tag{21}$$

REFERENCES

[1] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*. John Wiley & Sons, 2001.

[2] M. Bassevill and I. Nikiforov. *Digital Signal Processing: Detection of Abrupt Changes*. Prentice Hall, Englewood Cliffs, NJ, 1993.

[3] D. Bertsekas, A. Nedic, and A. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific, 2003.

[4] H.A.P. Blom and Y. Bar-Shalom. The interacting multiple model algorithm for systems with markovian switching coefficients. *Automatic Control, IEEE Transactions on*, 33(8):780 –783, aug 1988.

[5] J. Borwein and A. Lewis. *Convex Analysis and Nonlinear Optimization: Theory and Examples*. Canadian Mathematical Society, 2005.

[6] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, March 2004.

[7] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52:489–509, February 2006.

[8] E. Candès, M. Wakin, and S. Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier Analysis and Applications, special issue on sparsity*, 14(5):877–905, December 2008.

[9] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, April 2006.

[10] M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control*, volume 371/2008, pages 95–110. Springer Berlin / Heidelberg, 2008.

[11] M. Grant, S. Boyd, and Y. Ye. CVX: Matlab Software for Disciplined Convex Programming, June 2009.

[12] F. Gustafsson. *Adaptive Filtering and Change Detection*. Wiley, New York, 2001.

[13] Fredrik Gustafsson. *Statistical Sensor Fusion*. Studentlitteratur AB, 2010.

[14] Fredrik Gustafsson and S. F. Graebe. Closed loop performance monitoring in the presence of system changes and disturbances. *Automatica*, 34(11):1311–1326, 1998.

[15] S. Hui and S.H. Zak. Observer design for systems with unknown imputs. *Int. J. Appl. Math. Comput. Sci.*, 15(4), 2005.

[16] T. Kailath, A. Sayed, and B. Hassibi. *Linear Estimation*. Information and System Sciences. Prentice-Hall, Englewood Cliffs, NJ, 2000.

[17] R.E. Kalman. A new approach to linear filtering and prediction problems. *J Basic Engr. Trans. ASME Series D*, 82:35–45, 1960.

[18] S.-J. Kim, K. Koh, S. Boyd, and D. Gorinevsky. ℓ_1 trend filtering. *SIAM Review*, 51(2):339–360, 2009.

[19] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. An interior-point method for large-scale l1-regularized least squares. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):606–617, December 2007.

[20] L. Ljung. *System Identification - Theory for the User*. Prentice-Hall, Upper Saddle River, N.J., 2nd edition, 1999.

[21] L. Ljung and T. Kailath. A unified approach to smoothing formulas. *Automatica*, 12(2):147–157, 1976.

[22] J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[23] Henrik Ohlsson, Lennart Ljung, and Stephen Boyd. Segmentation of ARX-models using sum-of-norms regularization. *Automatica*, 46(6):1107 – 1111, 2010.

[24] E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954.

[25] R. Patton, P. Frank, and R. Clark. *Fault diagnosis in dynamic systems*. Prentice Hall, 1989.

[26] R. Rockafellar. *Convex Analysis*. Princeton University Press, 1996.

[27] J. Roll. Piecwise linear solution paths with application to direct weight optimizatiom. *Automatica*, 44:2745–2753, 2008.

[28] S. Särkkä and J. Hartikainen. EKF/UKF toolbox for Matlab 7.x, November 2007. Version 1.2.

[29] R. Tibsharani. Regression shrinkage and selection via the lasso. *J Roy Statistical Society B (Methodological)*, 58(1):267–288, 1996.

[30] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68:49–67, 2006.