Linear system identification using stable spline kernels and PLQ penalties

Aleksandr Y. Aravkin, James V. Burke and Gianluigi Pillonetto

***Abstract*— The classical approach to linear system identification is given by parametric Prediction Error Methods (PEM). In this context, model complexity is often unknown so that a model order selection step is needed to suitably trade-off bias and variance. Recently, a different approach to linear system identification has been introduced, where model order determination is avoided by using a regularized least squares framework. In particular, the penalty term on the impulse response is defined by so called *stable spline kernels*. They embed information on regularity and BIBO stability, and depend on a small number of parameters which can be estimated from data. In this paper, we provide new nonsmooth formulations of the stable spline estimator. In particular, we consider linear system identification problems in a very broad context, where regularization functionals and data misfits can come from a rich set of piecewise linear quadratic functions. Moreover, our analysis includes polyhedral inequality constraints on the unknown impulse response. For any formulation in this class, we show that interior point methods can be used to solve the system identification problem, with complexity $O(n^3)+O(mn^2)$ in each iteration, where n and m are the number of impulse response coefficients and measurements, respectively. The usefulness of the framework is illustrated via a numerical experiment where output measurements are contaminated by outliers.**

***Index Terms*— linear system identification; bias-variance trade off; kernel-based regularization; robust statistics; interior point methods; piecewise linear quadratic densities**

I. Introduction

The classical approach to linear system identification is given by Parametric Prediction Error Methods (PEM) [1], [2]. First, models of different and unknown order, e.g. ARX or ARMAX, are postulated and identified from data. Then, they are compared using either complexity measures such as AIC or cross validation (CV) [3], [4].

Some limitations of this approach have been recently described in [5] (see also [6] for an analysis of CV). This has led to the introduction of an alternative technique, where identification is seen as a function learning problem formulated in a possibly infinite-dimensional space [5], [7]. In particular, the problem is cast in the framework of Gaussian regression [8]: the unknown impulse response is seen as a Gaussian process, whose autocovariance encodes available prior knowledge. This approach was subsequently given an interpretation in a Regularized Least Squares framework in [9].

The new estimators proposed in [5], [10] rely on a class of autocovariances, called stable spline kernels, which include information on the exponential stability of the unknown system. The impulse response is modeled as the m-fold integration of white Gaussian noise subject to an exponential time transformation. The first-order stable spline kernel has been recently derived using *deterministic* arguments [9], and named the TC kernel. An even more sophisticated covariance for system identification, the so called DC kernel, is also described in [9].

All of these kernels are defined by a small number of unknown hyperparameters, which can be learned from data, e.g. by optimizing the marginal likelihood [11], [12], [13]. This procedure resembles model order selection in the classical parametric paradigm, and theoretical arguments supporting it are illustrated in [14]. Once the hyperparameters are found, the estimate of the system impulse response becomes available in closed form. Extensive simulation studies have shown that these new estimators can lead to significant advantages with respect to the classical ones, in particular in terms of robustness and in model complexity selection.

All of the new kernel-based approaches discussed in [5], [7], [9] rely on quadratic loss and and penalty functions. As a result, in some circumstances they may perform poorly. In fact, quadratic penalties are not robust when outliers are present in the data [15], [16], [17], [18]. In addition, they neither promote sparse solutions, nor select small subsets of measurements or impulse response coefficients with the greatest impact on the predictive capability for future data. These are key issues for feature selection and compressed sensing [19], [20], [21].

The limitations of quadratic penalties motivate adopting alternative penalties for both loss and regularization functionals. For example, popular regularizers are the the ℓ_1-norm, as in the LASSO [22], or a weighted combination of ℓ_1 and ℓ_2, as in the elastic net procedure [23]. Popular fitting measures robust to outliers are the ℓ_1-norm, the Huber loss [15], the Vapnik ε-insensitive loss [24], [25] and the hinge loss [26], [25], [27]. Recently, all of these approaches have been cast in a unified statistical modeling framework [14], [28], where solutions to all models can be computed using interior point (IP) methods.

The aim of this paper is to extend this framework to the linear system identification problem. In particular, we propose new impulse response estimators that combine the

A.Y. Aravkin (saravkin@us.ibm.com) is with IBM T.J. Watson Research Center, Yorktown Heights, NY, 10598

J.V. Burke (burke@math.washington.edu) is with Department of Mathematics, University of Washington, Seattle, USA

G. Pillonetto (giapi@dei.unipd.it) is with Dipartimento di Ingegneria dell'Informazione, University of Padova, Padova, Italy.

This research has been partially supported by the European Community under agreement n. FP7-ICT-223866-FeedNetBack, n257462 HYCON2 Network of excellence, by the FIRB project entitled "Learning meets time", and by the advanced grant LEARN from the European Research Council, contract 267381.

stable spline kernels and arbitrary piecewise linear quadratic (PLQ) penalties. Generalizing the work in [14], [28], we also allow the inclusion of inequality constraints on the unknown parameters. This generalization can be used to efficiently include additional information — for example, about nonnegativity and unimodality of the impulse response — into the final estimate. We show that all of these models can be solved with IP techniques, with complexity that scales well with the number of output measurements. These new identification procedures are tested via a Monte Carlo study where output error models are randomly generated and output data (corrupted by outliers) is obtained. We compare the performance of the classical stable spline estimator that uses a quadratic loss with the performance of the new estimator that uses ℓ_1 loss.

The structure of the paper is as follows. In Section II, we formulate the problem and briefly review the stable spline estimator described in [5], [9]. In Section III we introduce the new class of non smooth stable spline estimators, review the class of PLQ penalties, and generalize the framework in [29] by including affine inequality constraints. We also demonstrate how IP methods can be used to efficiently compute the impulse response estimates. In Section IV, the new approach is tested via a Monte Carlo study, where system output measurements are corrupted by outliers. We end the paper with Conclusions, and include additional proofs in the Appendix.

II. Problem statement and the stable spline estimator

A. Statement of the problem

Consider the following linear time-invariant discrete-time system

$$y(t) = G(q)u(t) + e(t), \quad t = 1, \dots, m\,, \tag{II.1}$$

where y is the output, q is the shift operator ($qu(t) = u(t+1)$), $G(q)$ is the linear operator associated with the true system, assumed stable, u the input and e the i.i.d. noise. Assuming the input u known, our problem is to estimate the system impulse response from N noisy measurements of y.

B. The stable spline estimator

We now briefly review the regularized approach to system identification proposed in [5], [9]. For this purpose, denote by $x \in \mathbb{R}^n$ the (column) vector containing the impulse response coefficients. Here, in contrast to classical approaches to system identification, the size n is chosen sufficiently large to capture system dynamics rather than to establish any kind of trade-off between bias and variance. It is useful to rewrite the measurement model (II.1) using the following matrix-vector notation

$$z = Hx + E\,, \tag{II.2}$$

where the vector $z \in \mathbb{R}^m$ contains the m output measurements, H is a suitable matrix defined by input values, and E denotes the noise of unknown variance σ^2. Then, the stable spline estimator is defined by the following regularized least squares problem:

$$\hat{x} = \arg\min_x \|z - Hx\|_2^2 + \gamma x^T Q^{-1} x\,, \tag{II.3}$$

where the positive scalar γ is a regularization parameter, and $Q \in \mathbb{R}^{n\times n}$ can be taken from the class of stable spline kernels [10]. In particular, adopting the discrete-time version of the stable spline kernel of order 1, the (i,j) entry of Q is

$$Q_{ij} = \alpha^{\max(i,j)}, \quad 0 \le \alpha < 1\,. \tag{II.4}$$

Above, α is a kernel hyperparameter which corresponds to the dominant pole of the system, and is typically unknown. This kernel was also studied in [9], where it was called the tuned/correlated (TC) kernel. Motivations underlying the particular shape (II.4) have been discussed under both a statistical and a deterministic framework, see [30] and [31]. Note that the estimator (II.3), equipped with the kernel (II.4), contains the unknown hyperparameters α and γ. These can be obtained as follows. First, the estimate $\hat{\sigma}^2$ of σ^2 can be computed by fitting a low-bias model for the impulse response using least squares (as e.g. described in [32]). Then, one can exploit the Bayesian interpretation underlying problem (II.3): if the noise is Gaussian, it provides the minimum variance estimate of x when the impulse response is modeled as a Gaussian vector independent of E with autocovariance λQ. Here, λ is an unknown scale factor equal to σ^2/γ. The estimates of λ and α are obtained by maximizing the marginal likelihood (obtained by integrating x out of the joint density of z and x). This gives

$$(\hat{\lambda}, \hat{\alpha}) = \arg\min_{\lambda,\alpha} z^T \Sigma_z^{-1} z + \log\det(\Sigma_z)\,, \tag{II.5}$$

where the $m \times m$ matrix Σ_z is

$$\Sigma_z = \lambda H Q H^T + \hat{\sigma}^2 I_m\,,$$

and I_m the $m \times m$ identity matrix (see [5] for details).

Let $\hat{Q}$ be the matrix defined in (II.4) with α set to its estimate $\hat{\alpha}$. Then, setting Q to $\hat{Q}$ and γ to $\hat{\sigma}^2/\hat{\lambda}$ in (II.3), we obtain the impulse response estimate

$$\hat{x} = \hat{\lambda}\hat{Q}H^T\hat{\Sigma}_z^{-1} z\,,$$

where

$$\hat{\Sigma}_z = \hat{\lambda} H \hat{Q} H^T + \hat{\sigma}^2 I_m.$$

III. New non smooth formulations of the stable spline estimator

To simplify the problem formulation, it is useful to introduce an auxiliary variable y, and to to rewrite the classical stable spline estimator (II.3) using the following relationships:

$$x = Ly, \quad Q = LL^T. \tag{III.1}$$

where L is invertible. Using (III.1), (II.3)) becomes

$$\min_y \|(z - HLy)\|^2 + \gamma\|y\|^2\,. \tag{III.2}$$

It is apparent that this estimator uses quadratic functions to define both the loss $\|(z - HLy)\|^2$ and the regularizer $\|y\|^2$.

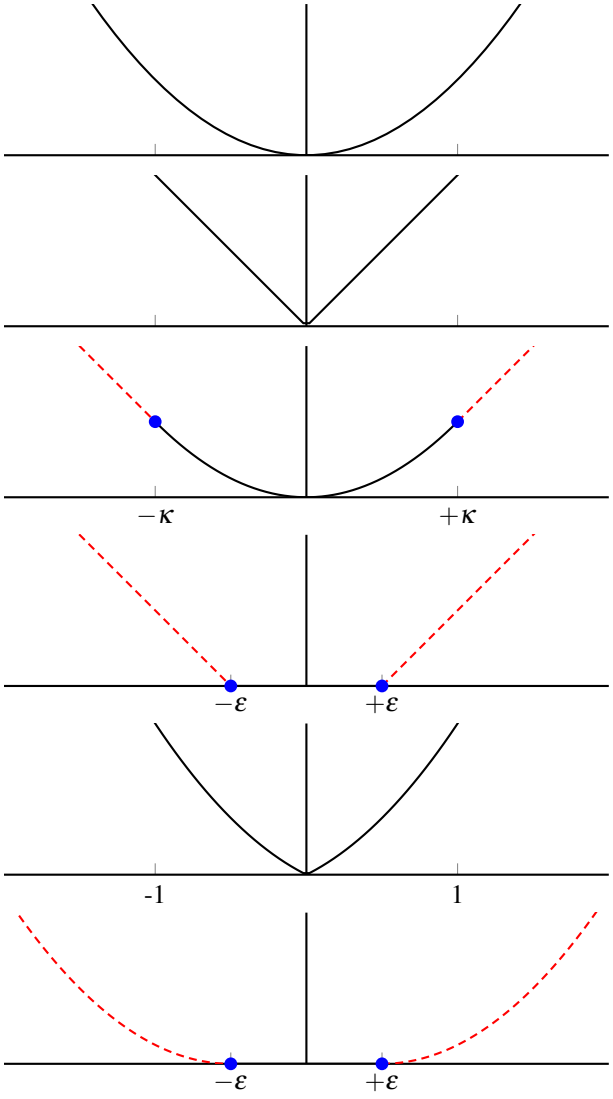

Fig. 1. Scalar penalties, top to bottom: ℓ_2, ℓ_1, Huber, Vapnik, elastic net, and smooth insensitive loss

In the rest of the paper we study a generalization of (III.2) given by

$$\min_{y\in Y} \quad V(HLy-z)+\gamma W(y)\ , \tag{III.3}$$

where Y is a polyhedral set (which can be used e.g. to provide nonnegativity information on the impulse response $x=Ly$), and V, W are defined by the piecewise linear quadratic functions introduced in the next subsection.

A. PLQ penalties

Definition 3.1 (PLQ functions and penalties): A piecewise linear quadratic (PLQ) function is any function $\rho(U,M,b,B;\cdot):\mathbb{R}^N\to\overline{\mathbb{R}}$ having representation

$$\rho(U,M,b,B;y)=\sup_{u\in U}\left\{\langle u,b+By\rangle-\tfrac{1}{2}\langle u,Mu\rangle\right\}\ , \tag{III.4}$$

where $U\subset\mathbb{R}^K$ is a nonempty polyhedral set, $M\in\mathscr{S}_+^K$ the set of real symmetric positive semidefinite matrices, and $b+By$ is an injective affine transformation in y, with $B\in\mathbb{R}^{K\times N}$, so, in particular, $K\geq N$ and $\text{null}(B)=\{0\}$.

When $0\in U$, the associated function is a *penalty*, since it is necessarily non-negative.

Remark 3.2: When $b=0$ and $B=I$, we recover the basic piecewise linear-quadratic penalties characterized in [33, Example 11.18].

Remark 3.3 (scalar examples): ℓ_2, ℓ_1, elastic net, Huber, hinge, and Vapnik penalties are all representable using the notation of Definition 3.1.

1) ℓ_2: Take $U=\mathbb{R}$, $M=1$, $b=0$, and $B=1$. We obtain
$$\rho(y)=\sup_{u\in\mathbb{R}}\left\{uy-u^2/2\right\}\ .$$
The function inside the sup is maximized at $u=y$, hence $\rho(y)=\frac{1}{2}y^2$.
2) ℓ_1: Take $U=[-1,1]$, $M=0$, $b=0$, and $B=1$. We obtain
$$\rho(y)=\sup_{u\in[-1,1]}\{uy\}\ .$$
The function inside the sup is maximized by taking $u=\mathbb{R}sign(y)$, hence $\rho(y)=|y|$.
3) Elastic net: $\ell_2+\lambda\ell_1$. Take
$$U=\mathbb{R}\times[-\lambda,\lambda],\ b=\begin{bmatrix}0\\0\end{bmatrix},\ M=\begin{bmatrix}1&0\\0&0\end{bmatrix},\ B=\begin{bmatrix}1\\1\end{bmatrix}\ .$$
4) Huber: Take $U=[-\kappa,\kappa]$, $M=1$, $b=0$, and $B=1$. We obtain
$$\rho(y)=\sup_{u\in[-\kappa,\kappa]}\left\{uy-u^2/2\right\}\ ,$$
with three explicit cases:
   a) If $y<-\kappa$, take $u=-\kappa$ to obtain $-\kappa y-\frac{1}{2}\kappa^2$.
   b) If $-\kappa\leq y\leq\kappa$, take $u=y$ to obtain $\frac{1}{2}y^2$.
   c) If $y>\kappa$, take $u=\kappa$ to obtain a contribution of $\kappa y-\frac{1}{2}\kappa^2$.

   This is the Huber penalty.
5) Vapnik loss is given by $(y-\varepsilon)_+ + (-y-\varepsilon)_+$. We obtain its PLQ representation by taking
$$B=\begin{bmatrix}1\\-1\end{bmatrix},\ b=-\begin{bmatrix}\varepsilon\\\varepsilon\end{bmatrix},\ M=\begin{bmatrix}0&0\\0&0\end{bmatrix},\ U=[0,1]\times[0,1]$$
to yield
$$\rho(y)=\sup_{u\in U}\left\{\left\langle\begin{bmatrix}y-\varepsilon\\-y-\varepsilon\end{bmatrix},u\right\rangle\right\}=(y-\varepsilon)_+ +(-y-\varepsilon)_+.$$
6) Soft insensitive loss function [34]. We can create a symmetric soft insensitive loss function (which one might term the Hubnik) by adding together two soft hinge loss functions:
$$\begin{aligned}\rho(y)&=\sup_{u\in[0,\kappa]}\{(y-\varepsilon)u\}-\tfrac{1}{2}u^2+\sup_{u\in[0,\kappa]}\{(-y-\varepsilon)u\}-\tfrac{1}{2}u^2\\&=\sup_{u\in[0,\kappa]^2}\left\{\left\langle\begin{bmatrix}y-\varepsilon\\-y-\varepsilon\end{bmatrix},u\right\rangle\right\}-\tfrac{1}{2}u^T\begin{bmatrix}1&0\\0&1\end{bmatrix}u\ .\end{aligned}$$
See bottom bottom panel of Fig. 1.

B. Optimization with PLQ penalties

Consider a *constrained* minimization problem for a general PLQ penalty:

$$\min_{y\in Y} \quad \rho_{U,M,b,B}(y) := \sup_{u\in U}\left\{\langle u, b+By\rangle - \frac{1}{2}u^T Mu\right\}, \quad \text{(III.5)}$$

where Y is a polyhedral set, described by

$$Y = \{y : A^T y \le a\}. \quad \text{(III.6)}$$

After studying this problem, we will come back to consider the estimator (III.3).

It turns out that a wide class of problems (III.5) are solvable by interior point (IP) methods [35], [36], [37]. IP methods solve nonsmooth optimization problems by working directly with smooth systems of equations characterizing the optimality of these problems. [29, Theorem 13] presents a full convergence analysis for IP methods for formulations (III.5) without inequality constraints, so $Y = \mathbb{R}^N$in (III.5). While a generalization of the full analysis to cover inequality constraints is out of the scope of this paper, we present an important computational result showing that constraints can be included in a straightforward manner, and provide the computational complexity of each interior point iteration. Moreover, the proof of the result (given in Appendix) shows that constraints *help the numerical stability* of the interior point iterations.

Theorem 3.4 (Interior Point for PLQ with Constraints): Consider any optimization problem of the form (III.5), with $y \in \mathbb{R}^N$, $b,u \in \mathbb{R}^K$, $C \in \mathbb{R}^{K\times L}$, $c \in \mathbb{R}^L$, $B \in \mathbb{R}^{K\times N}$, $A \in \mathbb{R}^{N\times P}$, $M \in \mathbb{R}^{K\times K}$, and $a \in \mathbb{R}^P$. Suppose that the PLQ satisfies

$$\text{Null}(M) \cap \text{Null}(C^T) = 0. \quad \text{(III.7)}$$

Suppose also that M contains on the order of K entries, while C contains on the order of L entries. Then every interior point iterations can be computed with complexity $O(L + KN^2 + PN^2 + N^3)$.

The assumptions on the structure of M and C are satisfied for many common PLQ penalties. For example, for ℓ_2 we have $M = I$ and $C = 0$, while for ℓ_1, $M = 0$ and C contains two copies of the identity matrix.

Turning out attention back to system identification, $N = n$ will be the dimension of the impulse response, while K and L may depend on m; in fact $K \ge m$ always, while L depends on the structure of the PLQ penalty. To be more specific, we have the following corollary.

Corollary 3.5: Problem (III.3) can be formulated as a minimization problem of the form (III.5). If the constraint matrix A has on the order of n entries, while matrices B and C have on the order of m entries, each interior point iteration can be solved with complexity $O(mn^2 + n^3)$.

Note that the computational complexity of the IP method scales favorably with the number of measurements m which, in the system identification scenario, is typically much larger than the number of unknown impulse response coefficients n.

IV. Monte Carlo study

We consider a Monte Carlo study of 300 runs. At each run, the MATLAB command `m=rss(30)` is first used to obtain a SISO continuous-time system of 30th order. The continuous-time system `m` is then sampled at 3 times of its bandwidth, obtaining the discrete-time system `md` through the commands: `bw=bandwidth(m); f = bw*3*2*pi; md=c2d(m,1/f,'zoh')`. If all poles of `md` are within the circle with center at the origin and radius 0.95 on the complex plane, then the feedforward matrix of `md` is set to 0, i.e. `md.d=0`, and the system is used and saved.

The system input at each run is white Gaussian noise of unit variance. The input delay is always equal to 1 and this information is given to every estimator used in the Monte Carlo study described below.

Data consists of 400 input-output pairs, which are collected after getting rid of initial conditions, and corrupted by a noise generated as a mixture of two normals with a fraction of outlier contamination equal to 0.2; i.e.,

$$e_i \sim 0.8\mathbf{N}(0,\sigma^2) + 0.2\mathbf{N}(0,100\sigma^2).$$

Here, σ^2 is randomly generated in each run as the variance of the noiseless output divided by the realization of a random variable uniformly distributed on $[1,10]$. With probability 0.2, each measurement may be contaminated by a random error whose standard deviation is 10σ.

The quality of an estimator is measured by computing the fit measure at every run. To be more specific, given a generic dynamic system represented by $S(q)$, let $\|S(q)\|_2$ denote the ℓ_2 norm of its impulse response, numerically computed using only the first 100 impulse response coefficients, whose mean is denoted by $\bar{S}(q)$. Then, the fit measure for the j-th run with estimated model $\hat{G}_j(q)$ is

$$\mathscr{F}_j(G,\hat{G}_j) = 100\left(1 - \frac{\|G(q)-\hat{G}_j(q)\|_2}{\|G(q)-\bar{G}(q)\|_2}\right) \quad \text{(IV.1)}$$

During the Monte Carlo simulations, the following 5 estimators are used:

- *Oe+oracle*. Classical PEM approach, with candidate models given by rational transfer functions defined by two polynomials of the same order. This estimator is implemented using the `oe.m` function of the MATLAB System Identification Toolbox equipped with the robustification option (`'LimitError',r`)[1] and an oracle, which provides a bound on the best achievable performance of PEM by selecting (at every run) the model order (between 1 and 20) and the value of `r` (0, 1, 2 or 3) that maximize (IV.1).
- *Oe+CV*. Same as above, except that `r=0` (the fit criterion is purely quadratic) and model order is estimated

[1] As per MATLAB documentation, the value of `r` specifies when to adjust the weight of large errors from quadratic to linear. Errors larger than `r` times the estimated standard deviation have a linear weight in the criteria. The standard deviation is estimated robustly as the median of the absolute deviations from the median and divided by 0.7. The value `r=0` disables the robustification and leads to a purely quadratic criterion.

via cross validation. In particular, data are split into a training and validation data set of equal size. Then, for every model order ranging from 1 to 20, the MATLAB function `oe.m` (fed with the training set) is called. The estimate of the order minimizes the sum of squared prediction errors on the validation set. This is obtained by the MATLAB function `predict.m` (imposing null initial conditions) fed with the validation data set. The final model is computed by `oe.m`, using the estimated value of the order and all the available measurements (the union of the training and validation sets).

- *Oe+CVrob*. Same as above, except that level of robustification `r` is also chosen via cross validation on the grid $\{0,1,2,3\}$.
- *SS+ℓ_2*. This is the classical stable spline estimator (II.3), which uses a quadratic loss and the stable spline regularizer. Hyperparameters are determined via marginal likelihood optimization, as described in subsection II-B. The number of estimated impulse response coefficients, i.e. the dimension of x in (II.2), is $n = 100$. Only the first 100 input-output pairs are used to define the entries of the matrix H in (II.2), so that the size of the measurement vector z is $m = 300$.
- *SS+ℓ_1*. This is the new nonsmooth version of the stable spline estimator. It coincides with (II.3) except that the quadratic loss is replaced by the ℓ_1 loss. The hyperparameter α defining the stable spline kernel in (II.4) and the regularization parameter γ are estimated via cross validation as follows. The matrix H in (II.2) is defined as described above. Then, the remaining 300 input-output pairs are split into a training and validation data set of equal size. The estimates of the hyperparameters α, γ are chosen so that the corresponding impulse response estimate (obtained using only the training set) provides the best prediction on the validation data (according to a quadratic fit). The candidate hyperparameters are selected from a two-dimensional grid. In particular, α may assume values in $[0.01, 0.05, 0.1, 0.15, \ldots, 0.9, 0.95, 0.99]$ while γ varies on a set given by 20 values logarithmically spaced between $\hat{\gamma}/100$ and $100\hat{\gamma}$, where $\hat{\gamma}$ is the estimate used by *SS+ℓ_2*. The final estimate of the impulse response is computed using the hyperparameter estimates and the union of training and validation data sets.

The plots in Fig. 2 are the Matlab boxplots of the errors (IV.1) obtained by the 5 estimators. The rectangle shows the $25-75\%$ quantiles of all the numbers with the horizontal line showing the median. The "whiskers" outside the rectangle display the $10-90\%$ quantiles, with the remaining errors (which may be deemed outliers) plotted using "+". The average fits obtained by *Oe+oracle*, *Oe+CV*, *Oe+CVrob*, *SS+ℓ_2* and *SS+ℓ_1* are $84.7, 44.4, 62.6, 55.8$ and 70.1, respectively. The best results are obtained by *Oe+oracle*. However, keep in mind that this estimator relies on an ideal tuning of the model order and of the level of robustification which is not implementable in practice.

In comparison with the other estimators, the performance of *SS+ℓ_2* and *Oe+CV* is negatively influenced by the presence of data contamination. The reason is that both of these estimators use quadratic loss functions. Notice however that textitSS+ℓ_2 largely outperforms *Oe+CV*.
Focusing now on numerical schemes equipped with robust losses, we see that *SS+ℓ_1* outperforms *Oe+CVrob*. It provides the best results among all the estimators implementable in practice: the stable spline kernel introduces a suitable regularization with the ℓ_1 loss to guard against outliers.

V. Conclusions

We have extended the stable spline estimator to a non smooth setting. Quadratic losses and regularizers can now be replaced by general PLQ functions, which allow new applications, such as robust estimators in the presence of outliers in the data. In addition, we presented an extended formulation that can include affine inequality constraints on the unknown impulse response, which can be used for example to incorporate nonnegativity into the estimate. We have shown that the corresponding generalized estimates can be computed in an efficient way by IP methods. Finally, our simulation results showed a significant performance improvement of the stable spline kernel with ℓ_1 loss over previous art.

VI. Appendix

A. Proof of Theorem 3.4

From [33][Example 11.47], the Lagrangian for problem (III.5) for feasible (y,u) is given by

$$L(y,u) = b^T u - \frac{1}{2} u^T M u + u^T B y \, .$$

Since U is by assumption a polyhedral set, it can be expressed by a linear system of inequalities:

$$U = \{u : C^T u \leq c\} \, . \tag{VI.1}$$

Using the explicit characterizations of U and W, the optimality conditions for (III.5) are

$$\begin{aligned} 2By - Mu + b &= Cq \, , \; q \geq 0 \\ -B^T u &= Aw \, , \; w \geq 0 \end{aligned} \tag{VI.2}$$

(see [33] and [38] for more details). The inequality constraint in the definition of U in (VI.1) can be reformulated using slack variables s, r:

$$\begin{aligned} C^T u + s &= c \\ A^T y + r &= a \, . \end{aligned}$$

Combining all of these equations yields the KKT system for (III.5):

$$\begin{aligned} 0 &= B^T u + Aw \\ 0 &= By - Mu - Cq + b \\ 0 &= C^T u + s - c \\ 0 &= A^T y + r - a \\ 0 &= q_i s_i \; \forall i \, , \; q, s \geq 0 \\ 0 &= w_i r_i \; \forall i \, , \; w, r \geq 0 \, . \end{aligned} \tag{VI.3}$$

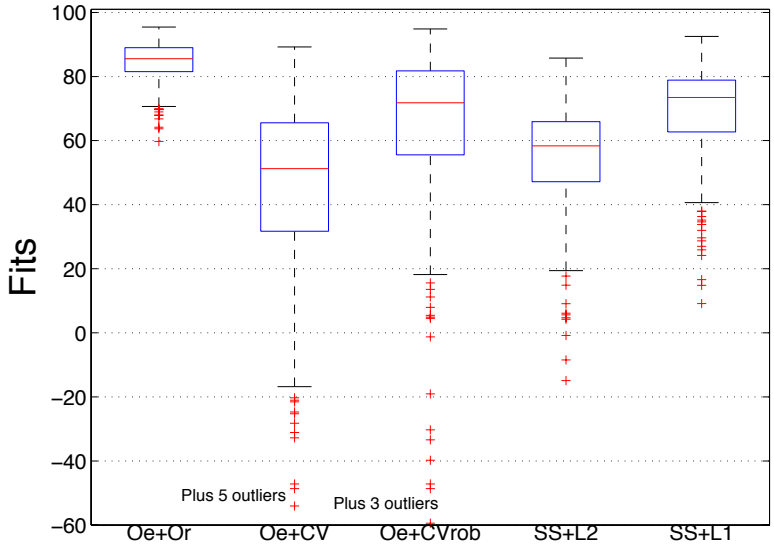

Fig. 2. Boxplot of the 300 percentage fits obtained by PEM equipped with an oracle (*Oe+Or*), by PEM with cross validation equipped with the quadratic loss (*Oe+CV*) and with a robust loss (*Oe+CVrob*), by the stable spline estimator equipped with the quadratic loss (*SS+*ℓ_2) and with the ℓ_1 loss (*SS+*ℓ_1).

The last two sets of equations in (VI.3) are known as the complementarity conditions. Solving the problem (III.5) is then equivalent to satisfying (VI.3), and there is a vast optimization literature on working directly with the KKT system. In the Kalman filtering/smoothing application, interior point methods have been used to solve the KKT system (VI.3) in a numerically stable and efficient manner, see e.g. [39].

An interior point approach applies damped Newton iterations to a relaxed version of VI.3:

$$F_\mu(s,q,u,r,w,y) = \begin{bmatrix} s+C^Tu-c \\ QS\mathbf{1}-\mu\mathbf{1} \\ By-Mu-Cq+b \\ r+A^Ty-a \\ WR\mathbf{1}-\mu\mathbf{1} \\ B^Tu+Aw \end{bmatrix}. \qquad \text{(VI.4)}$$

The relaxation parameter μ is driven aggressively to 0 as the method proceeds. Every Newton iteration solves

$$F_\mu^{(1)}\left[\Delta s^T,\quad \Delta q^T,\quad \Delta u^T,\quad \Delta r^T,\quad \Delta w^T,\quad \Delta y^T,\right]^T =$$

$$-F_\mu(s,q,u,r,w,y)\,,$$

where

$$F_\mu^{(1)} = \begin{bmatrix} I & 0 & C^T & 0 & 0 & 0 \\ Q & S & 0 & 0 & 0 & 0 \\ 0 & -C & -M & 0 & 0 & B \\ 0 & 0 & 0 & I & 0 & A^T \\ 0 & 0 & 0 & W & R & 0 \\ 0 & 0 & B^T & 0 & A & 0 \end{bmatrix}. \qquad \text{(VI.5)}$$

Using the row operations

$$r_2 \leftarrow r_2 - Qr_1$$
$$r_3 \leftarrow r_3 + CS^{-1}r_2$$

we arrive at the system

$$\begin{bmatrix} I & 0 & C^T & 0 & 0 & 0 \\ 0 & S & -QC^T & 0 & 0 & 0 \\ 0 & 0 & -T & 0 & 0 & B \\ 0 & 0 & 0 & I & 0 & A^T \\ 0 & 0 & 0 & W & R & 0 \\ 0 & 0 & B^T & 0 & A & 0 \end{bmatrix}.$$

where $T = M + C\mathrm{diag}(q/s)C^T$. Note that this matrix is invertible if and only if the hypothesis (III.7) holds. If T is invertible, the row operations

$$r_6 \leftarrow r_4 + B^TT^{-1}B$$
$$r_5 \leftarrow r_5 - Wr_4$$
$$r_6 \leftarrow r_6 - AR^{-1}r_5$$

reduce the system to upper triangular form

$$\begin{bmatrix} I & 0 & C^T & 0 & 0 & 0 \\ 0 & S & -QC^T & 0 & 0 & 0 \\ 0 & 0 & -T & 0 & 0 & B \\ 0 & 0 & 0 & I & 0 & A^T \\ 0 & 0 & 0 & 0 & R & -WA^T \\ 0 & 0 & 0 & 0 & 0 & B^TT^{-1}B + AR^{-1}WA^T \end{bmatrix}.$$

Note that s, q, r, and w are componentwise positive (which holds for every nonzero μ), while B is injective (see Definition 3.1), hence $B^TT^{-1}B$ is a square matrix of full rank. The term $AR^{-1}WA^T$ is also positive semidefinite, and only serves to stabilize the inversion of the final term. Therefore, we can carry out Newton iterations on the μ-relaxed system, as claimed.

To show the computational complexity, we give the full interior point iteration, which is derived by applying the row operations used to obtain the upper triangular system to the right hand side $-F_\mu$, then solving for Δy, and back

substituting.

$$
\begin{aligned}
r_1 &= -s - C^T u + c \\
r_2 &= \mu \mathbf{1} + Q(C^T u - c) \\
r_3 &= -(By - Mu - Cq + b) + CS^{-1} r_2 \\
r_4 &= -(r + A^T y - a) \\
r_5 &= \mu \mathbf{1} + W(A^T y - a) \\
T &= M + CQS^{-1}C^T \\
r_6 &= -(B^T u + Aw) + B^T T^{-1} r_3 - AR^{-1} r_5 \\
\Omega &= B^T T^{-1} B + AR^{-1} W A^T \\
\Delta y &= \Omega^{-1} r_6 \\
\Delta w &= R^{-1}(r_5 + WA^T \Delta y) \\
\Delta r &= r_4 - A^T \Delta y \\
\Delta u &= T^{-1}(-r_3 + B\Delta y) \\
\Delta q &= S^{-1}(r_2 + QC^T \Delta u) \\
\Delta s &= r_1 - C^T \Delta u
\end{aligned}
\tag{VI.6}
$$

Note that the matrix T can be constructed in $O(L+K)$ operations if C contains on the order of L terms. The matrix Ω can be constructed in $O(NK^2 + NP^2)$ operations, and inverted in $O(N^3)$ operations. These operations dominate the complexity, giving the bound $O(L + NK^2 + NP^2 + N^3)$.

B. Proof of Corollary 3.5

To translate (III.3) to (III.5), we have to specify the structures A, B, b, C, c, which capture the impulse response constraints, the injective linear model, and the structure of U, respectively.

Suppose that $\rho_w(y)$ and $\rho_v(x)$ are given by

$$
\begin{aligned}
\rho_w(y) &:= \sup_{u \in U_w} \langle b_w + B_w y, u \rangle - \frac{1}{2} u^T M_w u \\
\rho_v(x) &:= \sup_{u \in U_v} \langle b_v + B_v x, u \rangle - \frac{1}{2} u^T M_v u
\end{aligned}
\tag{VI.7}
$$

First define

$$
\begin{aligned}
\widetilde{\rho}_v(y) &:= \rho_v(\gamma^{-1}(HLy - z)) \\
&= \sup_{u \in U_v} \left\langle b_v - \gamma^{-1} B_v z + \gamma^{-1} B_v HLy, u \right\rangle - \frac{1}{2} u^T M_v u \, .
\end{aligned}
$$

Adding $\widetilde{\rho}_v$ and ρ_w together, we obtain the general system identification objective with the following specification:

$$
M = \begin{bmatrix} M_w & 0 \\ 0 & M_v \end{bmatrix}, \quad B = \begin{bmatrix} B_w \\ \gamma^{-1} B_v HL \end{bmatrix}, \quad b = \begin{bmatrix} b_w \\ b_v - \gamma^{-1} B_v z \end{bmatrix}
$$
$$
C = \begin{bmatrix} C_w & 0 \\ 0 & C_v \end{bmatrix}, \quad c = \begin{bmatrix} c_w \\ c_v \end{bmatrix} \, .
$$

The matrix A and vector a encodes the constraints, as given by (III.6).

This completes the specification. The complexity result follows immediately from the assumptions on A, B, C and Theorem 3.4.

It is also worthwhile to consider the structure of (VI.6). First, note that

$$
\begin{aligned}
T &= M + CQS^{-1}C^T \\
&= \begin{bmatrix} M_w & 0 \\ 0 & M_v \end{bmatrix} + \begin{bmatrix} C_w & 0 \\ 0 & C_v \end{bmatrix} QS^{-1} \begin{bmatrix} C_w & 0 \\ 0 & C_v \end{bmatrix}^T \\
&= \begin{bmatrix} M_w + C_w Q_w S_W^{-1} C_w^T & 0 \\ 0 & M_v + C_v Q_v S^{-1} v C_v^T \end{bmatrix} \\
&= \begin{bmatrix} T_w & 0 \\ 0 & T_v \end{bmatrix} ,
\end{aligned}
$$

so in fact T is block diagonal. This fact gives a more explicit formula for Ω:

$$
\begin{aligned}
\Omega &= B^T T^{-1} B + AR^{-1} W A^T \\
&= \begin{bmatrix} B_w^T & \gamma^{-1} L^T H^T B_v^T \end{bmatrix} \begin{bmatrix} T_w^{-1} & 0 \\ 0 & T_v^{-1} \end{bmatrix} \begin{bmatrix} B_w \\ \gamma^{-1} B_v HL \end{bmatrix} + AR^{-1} W A^T \\
&= B_w^T T_w^{-1} B_w + \sigma^{-2} L^T H^T B_v^T T_v^{-1} B_v HL + AR^{-1} W A^T \, .
\end{aligned}
$$

References

[1] L. Ljung, *System Identification, Theory for the User*. Prentice Hall, 1999.

[2] T. Söderström and P. Stoica, *System Identification*. Prentice-Hall, 1989.

[3] H. Akaike, "A new look at the statistical model identification," *IEEE Transactions on Automatic Control*, vol. 19, pp. 716–723, 1974.

[4] T. J. Hastie, R. J. Tibshirani, and J. Friedman, *The Elements of Statistical Learning. Data Mining, Inference and Prediction*. Canada: Springer, 2001.

[5] G. Pillonetto and G. De Nicolao, "A new kernel-based approach for linear system identification," *Automatica*, vol. 46, no. 1, pp. 81–93, 2010.

[6] ——, "Pitfalls of the parametric approaches exploiting cross-validation or model order selection," in *Proceedings of the 16th IFAC Symposium on System Identification (SysId 2012)*, 2012.

[7] G. Pillonetto, A. Chiuso, and G. D. Nicolao, "Prediction error identification of linear systems: a nonparametric Gaussian regression approach," *Automatica*, vol. 47, no. 2, pp. 291–305, 2011.

[8] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[9] T. Chen, H. Ohlsson, and L. Ljung, "On the estimation of transfer functions, regularizations and Gaussian processes - revisited," *Automatica*, vol. 48, no. 8, pp. 1525–1535, 2012.

[10] G. Pillonetto, A. Chiuso, and G. De Nicolao, "Regularized estimation of sums of exponentials in spaces generated by stable spline kernels," in *Proceedings of the IEEE American Cont. Conf., Baltimora, USA*, 2010.

[11] J. S. Maritz and T. Lwin, *Empirical Bayes Method*. Chapman and Hall, 1989.

[12] D. MacKay, "Bayesian interpolation," *Neural Computation*, vol. 4, pp. 415–447, 1992.

[13] J. Berger, *Statistical Decision Theory and Bayesian Analysis*, 2nd ed., ser. Springer Series in Statistics. Springer, 1985.

[14] A. Aravkin, J. Burke, and G. Pillonetto, "A statistical and computational theory for robust and sparse kalman smoothing," in *Proceedings of the 16th IFAC Symposium on System Identification (SysId 2012)*, 2012.

[15] P. Huber, *Robust Statistics*. Wiley, 1981.

[16] J. Gao, "Robust l1 principal component analysis and its Bayesian variational inference," *Neural Computation*, vol. 20, no. 2, pp. 555–572, February 2008.

[17] A. Aravkin, B. Bell, J. Burke, and G. Pillonetto, "An ℓ_1-laplace robust kalman smoother," *Automatic Control, IEEE Transactions on*, vol. 56, no. 12, pp. 2898–2911, dec. 2011.

[18] S. Farahmand, G. Giannakis, and D. Angelosante, "Doubly robust smoothing of dynamical processes via outlier sparsity constraints," *IEEE Transactions on Signal Processing*, vol. 59, pp. 4529–4543, 2011.

[19] T. J. Hastie and R. J. Tibshirani, "Generalized additive models," in *Monographs on Statistics and Applied Probability*. London, UK: Chapman and Hall, 1990, vol. 43.

[20] B. Efron, T. Hastie, L. Johnstone, and R. Tibshirani, "Least angle regression," *Annals of Statistics*, vol. 32, pp. 407–499, 2004.

[21] D. Donoho, "Compressed sensing," *IEEE Trans. on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[22] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society, Series B.*, vol. 58, pp. 267–288, 1996.

[23] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society, Series B*, vol. 67, pp. 301–320, 2005.

[24] V. Vapnik, *Statistical Learning Theory*. New York, NY, USA: Wiley, 1998.

[25] M. Pontil and A. Verri, "Properties of support vector machines," *Neural Computation*, vol. 10, pp. 955–974, 1998.

[26] T. Evgeniou, M. Pontil, and T. Poggio, "Regularization networks and support vector machines," *Advances in Computational Mathematics*, vol. 13, pp. 1–150, 2000.

[27] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," *Neural Computation*, vol. 12, pp. 1207–1245, 2000.

[28] A. Aravkin, J. Burke, and G. Pillonetto, "Nonsmooth regression and state estimation using piecewise quadratic log-concave densities," in *Proceedings of the 51st IEEE Conference on Decision and Control (CDC 2012)*, 2012.

[29] A. Y. Aravkin, J. V. Burke, and G. Pillonetto, "Sparse/robust estimation and kalman smoothing with nonsmooth log-concave densities: Modeling,computation, and theory, 2013."

[30] G. Pillonetto and G. De Nicolao, "Kernel selection in linear system identification – part I: A Gaussian process perspective," in *Proceedings of CDC-ECC*, 2011.

[31] T. Chen, H. Ohlsson, G. Goodwin, and L. Ljung, "Kernel selection in linear system identification – part II: A classical perspective," in *Proceedings of CDC-ECC*, 2011.

[32] G. Goodwin, M. Gevers, and B. Ninness, "Quantifying the error in estimated transfer functions with application to model order selection," *IEEE Transactions on Automatic Control*, vol. 37, no. 7, pp. 913–928, 1992.

[33] R. Rockafellar and R. Wets, *Variational Analysis*. Springer, 1998, vol. 317.

[34] W. Chu, S. S. Keerthi, and C. J. Ong, "A unified loss function in bayesian framework for support vector regression," in *In Proceeding of the 18th International Conference on Machine Learning*, 2001, pp. 51–58.

[35] M. Kojima, N. Megiddo, T. Noma, and A. Yoshise, *A Unified Approach to Interior Point Algorithms for Linear Complementarity Problems*, ser. Lecture Notes in Computer Science. Berlin, Germany: Springer Verlag, 1991, vol. 538.

[36] A. Nemirovskii and Y. Nesterov, *Interior-Point Polynomial Algorithms in Convex Programming*, ser. Studies in Applied Mathematics. Philadelphia, PA, USA: SIAM, 1994, vol. 13.

[37] S. Wright, *Primal-dual interior-point methods*. Englewood Cliffs, N.J., USA: Siam, 1997.

[38] R. Rockafellar, *Convex Analysis*, ser. Priceton Landmarks in Mathematics. Princeton University Press, 1970.

[39] A. Aravkin, B. Bell, J. Burke, and G. Pillonetto, "Learning using state space kernel machines," in *Proc. IFAC World Congress 2011*, Milan, Italy, 2011.