An ℓ_1-Laplace Robust Kalman Smoother

Aleksandr Y. Aravkin, Bradley M. Bell, James V. Burke, and Gianluigi Pillonetto

***Abstract*—Robustness is a major problem in Kalman filtering and smoothing that can be solved using heavy tailed distributions; e.g., ℓ_1-Laplace. This paper describes an algorithm for finding the maximum *a posteriori* (MAP) estimate of the Kalman smoother for a nonlinear model with Gaussian process noise and ℓ_1-Laplace observation noise. The algorithm uses the convex composite extension of the Gauss–Newton method. This yields convex programming subproblems to which an interior point path-following method is applied. The number of arithmetic operations required by the algorithm grows linearly with the number of time points because the algorithm preserves the underlying block tridiagonal structure of the Kalman smoother problem. Excellent fits are obtained with and without outliers, even though the outliers are simulated from distributions that are not ℓ_1-Laplace. It is also tested on actual data with a nonlinear measurement model for an underwater tracking experiment. The ℓ_1-Laplace smoother is able to construct a smoothed fit, without data removal, from data with very large outliers.**

***Index Terms*—Interior point methods, Kalman filtering, Kalman smoothing, moving horizon estimation, robust statistics.**

I. Introduction

THE Kalman filter is best known as a system of recursive equations that finds the minimum variance state estimate for x_N in the following model: for $k = 1, \ldots, N$

$$\begin{aligned} x_k &= g_k(x_{k-1}) + w_k \\ z_k &= h_k(x_k) + v_k \end{aligned} \tag{1}$$

where g_k, h_k are known linear functions, w_k, v_k are mutually independent Gaussian random variables, and Q_k, R_k are the known covariances of w_k, v_k, respectively. However, in many applications, the dynamics may be nonlinear and more uncertain than in the model (1) (see for example [37]). The covariance matrices (Q_k or R_k) may be unknown, and the process or measurement noise (w_k or v_k) may not be Gaussian. A "robust" Kalman filter or smoother guards against some violation of the standard assumptions. It behaves well under nominal conditions (for example, Gaussian noise with known moments), and behaves acceptably when nominal conditions are violated. A 1985 review paper by Kassam and Poor surveys a wide assortment of articles and books on robust filters, with emphasis on "minimax" filters, which are designed to minimize the maximum possible value of a loss function related to filter performance [29] (also see [11], [19], [30], [35]). Other approaches to robust Kalman filtering are reviewed in the 1994 article by Schick and Mitter [42] where the measurement noise v_k has heavy tails (and g_k, h_k are linear). The authors divide the estimation approaches into three categories: Bayesian, nonparametric, and minimax. They review the portion of the minimax literature that uses influence bounding functions to obtain recursive approximate conditional mean estimators (e.g., [34]). In addition, they present a different approximation for the conditional mean (under the assumption that the distribution of v_k is known).

Again, in the case where g_k and h_k are linear, Masreliez and Martin [34] used the influence function approach (also called robust regression [23], [26], [33]) on the innovation $z_k - h_k(\hat{x}_{k|k-1})$, where $\hat{x}_{k|k-1} = g_k(\hat{x}_{k-1|k-1})$ is the estimate at time k given the estimate at the previous time $\hat{x}_{k-1|k-1}$. They used an upper bound as their variance estimate and achieved a robust Kalman filter. The influence function approach also appears in the more recent papers [16] and [13]. In this approach the state estimate at the kth time $\hat{x}_{k|k}$ is viewed as the regression of the state estimate at the previous time $\hat{x}_{k-1|k-1}$ together with the measurements at the current time z_k[16, eq. (15)]. The robust cost function ρ (its derivative is the influence function) is applied to the measurement residuals; i.e., $\rho[z_k - h_k(\hat{x}_{k-1|k-1})]$, and is equivalent to the Kalman filter when $\rho(u) = \|u\|^2$. These approaches can be viewed as extensions of M-estimators (defined by Huber [26]) to the Kalman filter setting.

In this paper, we study the case of non-Gaussian heavy tailed measurement noise v_k, nonlinear process functions g_k, and nonlinear measurement functions h_k. Heavy tailed densities occur in applications related to glint noise [25], air turbulence, and asset returns. Heavier tails are also a reasonable model for a contaminated normal distribution where "bad" measurements occur due to equipment malfunction, secondary noise sources, or other anomalies. In this, and even more general settings, the minimum variance estimate for the state sequence $\{x_1, \ldots, x_N\}$ can be approximated using stochastic simulation methods, such as Markov chain Monte Carlo or particle filtering, e.g., see [15], [20], [24], [27], [32], and [44]. These methods are very versatile, and can be applied to a wide range of situations where the

Manuscript received October 26, 2009; revised June 10, 2010; accepted March 03, 2011. Date of publication April 07, 2011; date of current version December 07, 2011. This work was supported in part by the National Science Foundation (NSF) underGrant DMS-0505712, NSF grant VIGRE, and the National Institute of Health (NIH) under Grant NIBIB 2 P41 EB01975. This work was also supported in part by the PRIN Project "Sviluppo di nuovi metodi e algoritmi per l'identificazione, la stima Bayesiana e il controllo adattativo e distribuito", by the Progetto di Ateneo CPDA090135/09 funded by the University of Padova, by the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement n. 257462 HYCON2 Network of excellence. Recommended by Associate Editor E. Weyer.

A. Y. Aravkin is with the Department of Earth and Ocean Sciences, University of British Columbia, Vancouver, V6T 1Z4 BC, Canada (e-mail: saravkin@eos.ubc.ca; aleksand@u.washington.edu).

B. M. Bell is with the Applied Physics Lab, University of Washington, Seattle, WA 98105-6698 USA (e-mail: bradbell@apl.washington.edu; bradbell@washington.edu).

J. V. Burke is with the Department of Mathematics, University of Washington, Seattle, WA 98195 USA (e-mail: burke@math.washington.edu).

G. Pillonetto is with Dipartimento di Ingegneria dell'Informazione, University of Padova, 35131 Padova, Italy (e-mail: giapi@dei.unipd.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2011.2141430

standard hypotheses in (1) fail to be satisfied. However, these techniques are very computationally intensive since they require Monte-Carlo and/or MCMC integration. In addition, they typically require a delicate tuning of proposal densities to improve their convergence rates. Moreover, detection of convergence is often heuristic and uncertain.

The approach taken here differs sharply from both stochastic simulation methods and previous research on robust filtering. In contrast to stochastic simulation methods, our approach is based on an optimization perspective associated with the computation of the maximum *a posteriori* (MAP) estimate of the state sequence $\{x_k\}$ given observations $\{z_k\}$. In contrast to previous robust filtering methods, we simultaneously optimize with respect to the entire state sequence $\{x_k\}$ and generalize to nonlinear process functions $\{g_k\}$ and measurement functions $\{h_k\}$ (here and below $\{u_k\}$ denotes the finite sequence $\{u_1, \ldots, u_N\}$). The result is the MAP estimate $\{\hat{x}_k\}$ of the state sequence $\{x_k\}$ given observations $\{z_k\}$ where particular heavy tailed distributions (ℓ_1-Laplace and Huber) model the measurement noise $\{v_k\}$. An advantage of an optimization approach is that first-order (and possibly second-order) necessary conditions provide a certificate of optimality and can be used to assess convergence. In the linear case, these conditions are also sufficient. However, in the nonlinear case, at best they only provide an assurance of local optimality. Nonetheless, optimization methods are numerically efficient and can be used to obtain better proposal densities for stochastic simulation methods, e.g., [38].

In the Gaussian linear case, the Rauch–Tung–Striebel (RTS) smoother [3] simultaneously optimizes with respect to the entire state sequence $\{x_k\}$. The algorithm in [4] extended this to nonlinear process functions $\{g_k\}$ and measurement functions $\{h_k\}$. We extend the method in [4] from Gaussian process and observational noise to observational noise with a robust log-concave density such as the ℓ_1-Laplace or Huber densities. Our extension preserves the block tridiagonal structure in the underlying linear algebra and scales linearly with the number of states.

Since we use log-concave densities, the optimization problem for the computation of the MAP estimate is in *convex composite form* [9], [10], i.e., the composition of a convex function with a smooth nonlinear function. By linearizing the nonlinear function within the convex function, we obtain a Gauss–Newton subproblem for the computation of the next iterate. Overall convergence is established using techniques from constrained nonlinear programming.

The algorithmic design is based on the Gauss–Newton methodology for convex composite optimization [9] rather than constrained nonlinear programming. This approach is ideally suited to log-concave densities. In Section III, we show that the associated Gauss–Newton subproblems are convex quadratic programs in the case of the ℓ_1-Laplace density and extended linear-quadratic programs in the case of the Huber density. Both cases are treated in Section IV where convex duality is used to show that a primal-dual path following algorithm for solving these subproblems preserves the underlying block tridiagonal structure. In addition, it is also shown that the convex composite convergence theory inspires subproblem initialization and termination rules that are particularly efficient. These rules make strong use of the duality information available in the primal-dual path following algorithm (see Remark 2). The proposed subproblem termination rule is linked to the initialization strategy and automatically adjusts the subproblem solution accuracy based on the magnitude of the certificate of optimality for the nonlinear smoothing problem. Further implementation details are given in Section V.

The paper proceeds as follows. The ℓ_1-Laplace distribution is introduced and the smoothing model is precisely defined in Section II. The MAP objective function, a Gauss–Newton approximation for this objective, and a convex quadratic program (QP) that represents this approximation appear in Section III. An interior point method for solving the QP is presented in Section IV. This method preserves the block tridiagonal structure of the standard Kalman filter, thus demonstrating that the computational effort scales linearly with the number of time points N. An algorithm that solves the MAP estimation problem, and convergence theory for the algorithm, are provided in Section V. The method presented in this paper is compared to other methods using simulated data in Section VI and is applied to real data used for underwater tracking in Section VII. The real data set is particularly good for testing because it includes independent position measurements that can be used to verify the tracking results.

II. Smoother Model With ℓ_1-Laplace Measurement Noise

Simultaneous optimization with respect to all time points was recently applied to the inequality constrained Kalman smoothing problem using an interior point method (see [6]). We use similar techniques to produce a robust smoother where the measurement noise $\{v_k\}$ comes from independent realizations of a multivariate distribution which we refer to as the ℓ_1-Laplace distribution. In the case where the covariance matrix is the identity, the negative log of the ℓ_1-Laplace (Gaussian) density is proportional to the ℓ_1 norm (ℓ_2 norm squared) of v_k. This differs from the standard multivariate Laplace distribution [31] whose negative log density is proportional to the ℓ_2 norm of v_k (again when the covariance matrix is the identity).

For $u \in \mathbf{R}^m$ we use the notation $\|u\|_1$ for the ℓ_1 norm of u; i.e., $\|u\|_1 = |u_1| + \cdots + |u_m|$. Define the multivariate ℓ_1-Laplace distribution with mean μ and covariance R as having the following density:

$$\mathbf{p}(v_k) = \det(2R)^{-1/2} \exp\left[-\sqrt{2}\left\|R^{-1/2}(v_k - \mu)\right\|_1\right] \quad (2)$$

where $R^{1/2}$ denotes a Cholesky factor of the positive definite matrix R; i.e., $R^{1/2}(R^{1/2})^T = R$. One can verify that this is a probability distribution with covariance R using the change of variables $u = R^{-1/2}(v_k - \mu)$. This distribution has been applied to robust estimation problems; e.g., [18, eq. 2.3]. A comparison of the Gaussian and Laplace distributions is displayed in Fig. 1. This comparison includes the densities, negative log densities, and influence functions, for both distributions.

We use the following general model for the underlying dynamic system: for $k = 1, \ldots, N$

$$\begin{aligned} x_k &= g_k(x_{k-1}) + w_k \\ z_k &= h_k(x_k) + v_k \end{aligned} \quad (3)$$

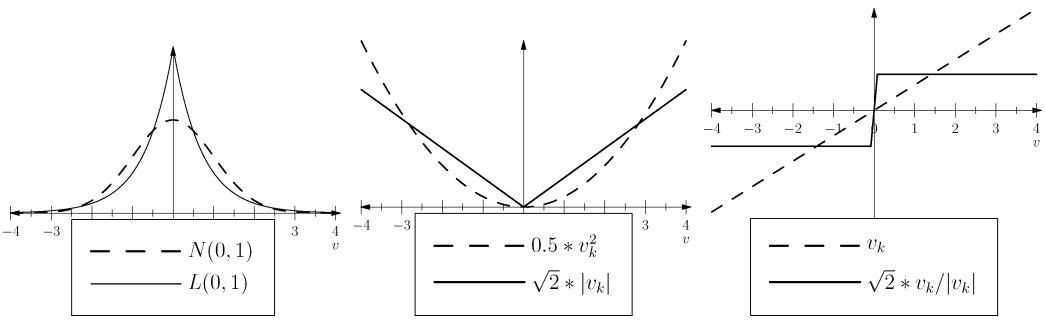

Fig. 1. Gaussian and Laplace densities, negative log densities, and influence functions (for scalar v_k).

where $g_k : \mathbf{R}^n \to \mathbf{R}^n$ is a known process function, $h_k : \mathbf{R}^n \to \mathbf{R}^{m(k)}$ is a known measurement function, the vector $w_k \in \mathbf{R}^n$ is unknown Gaussian process noise with mean zero and covariance $Q_k \in \mathbf{R}^{n\times n}$, and the vector $v_k \in \mathbf{R}^{m(k)}$ is unknown ℓ_1-Laplace measurement noise (2) with mean zero and covariance $R_k \in \mathbf{R}^{m(k)\times m(k)}$. Furthermore, we assume that the vectors $\{w_k\} \cup \{v_k\}$ are mutually independent. The initial state estimate is given by $g_1(x_0)$ which is constant; i.e., its derivative satisfies $g_1^{(1)}(x_0) = 0$.

III. Maximum *a Posteriori* Estimation

The MAP objective function is the negative log density corresponding to model (3); i.e.,

$$-\log \mathbf{p}(\{v_k\},\{w_k\}) = -\log \mathbf{p}(\{v_k\}) - \log \mathbf{p}(\{w_k\})\,. \quad (4)$$

Dropping terms that do not depend on $\{x_k\}$, maximizing the MAP objective with respect to $\{x_k\}$ is equivalent to minimizing

$$\sqrt{2}\sum_{k=1}^{N}\left\|R_k^{-1/2}[z_k - h_k(x_k)]\right\|_1 + \frac{1}{2}\sum_{k=1}^{N}[x_k - g_k(x_{k-1})]^T Q_k^{-1}[x_k - g_k(x_{k-1})]\,.$$

Given a sequence of column vectors $\{u_k\}$ and matrices $\{T_k\}$ we use the notation

$$\operatorname{vec}(\{u_k\}) = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix},\ \operatorname{diag}(\{T_k\}) = \begin{bmatrix} T_1 & 0 & \cdots & 0 \\ 0 & T_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & T_N \end{bmatrix}.$$

We also make the following definitions:

$$\begin{array}{ll} R = \operatorname{diag}(\{R_k\}) & x = \operatorname{vec}(\{x_k\}) \\ Q = \operatorname{diag}(\{Q_k\}) & \bar{w}(x) = \operatorname{vec}(\{x_k - g_k(x_{k-1})\}) \\ & \bar{v}(x) = \operatorname{vec}(\{z_k - h_k(x_k)\})\,. \end{array}$$

Using this notation, the MAP estimation problem is equivalent to

$$\underset{x\in\mathbf{R}^{Nn}}{\text{minimize}}\ \ \sqrt{2}\left\|R^{-1/2}\bar{v}(x)\right\|_1 + \frac{1}{2}\bar{w}(x)^T Q^{-1}\bar{w}(x). \quad (5)$$

This is an example of a convex composite optimization problem [9] wherein the objective function K takes the composite form $K = \rho \circ F$ with ρ convex and F smooth. The standard generalized Gauss–Newton approach to solving this problem is to linearize the smooth function F at the current iterate x^ν and then obtain the new iterate using the Gauss–Newton approximation

$$\rho[F(x)] \approx \rho\left[F(x^\nu) + F^{(1)}(x^\nu)(x - x^\nu)\right]$$

where $F^{(1)}$ is the derivative of F. The approach is typically coupled with either a line search or a trust-region strategy to ensure convergence far from a solution [9]. In the case of (5), the smooth function F has two components $\bar{v}$ and $\bar{w}$ that need to be linearized. We show that the resulting Gauss–Newton subproblem can be formulated as a structured convex quadratic program. To begin, reformulate (5) by introducing two new non-negative variables p^+ and p^- representing the positive and negative parts of $R^{-1/2}\bar{v}(x)$. This allows us to write the ℓ_1 portion of the objective as a linear expression in p^+ and p^- at the cost of adding non-negativity constraints for p^+ and p^- and a nonlinear equality constraint:

$$\begin{array}{ll} \text{minimize} & \sqrt{\mathbf{2}}^T(p^+ + p^-) + \frac{1}{2}\bar{w}(x)^T Q^{-1}\bar{w}(x) \\ \text{w.r.t} & x \in \mathbf{R}^{Nn},\ p^+ \in \mathbf{R}_+^M,\ p^- \in \mathbf{R}_+^M \\ \text{subject to} & R^{-1/2}\,\bar{v}(x) = p^+ - p^- \end{array}$$

where $M = m(1)+\cdots+m(N)$ is the total number of measurements, $\mathbf{R}_+$ is the non-negative real numbers, and $\sqrt{\mathbf{2}} \in \mathbf{R}^M$ is the vector with all its components equal to the square root of two. Using a representation similar to [6], define

$$\begin{aligned} S_k(x_{k-1}, x_k, p_k^+, p_k^-) &= \sqrt{2}\,1_{m(k)}^T\,(p_k^+ + p_k^-) \\ &\quad + \frac{1}{2}[x_k - g_k(x_{k-1})]^T Q_k^{-1}[x_k - g_k(x_{k-1})] \\ S(x, p^+, p^-) &= \sum_{k=1}^{N} S_k(x_{k-1}, x_k, p_k^+, p_k^-) \end{aligned}$$

where $p_k^+ \in \mathbf{R}_+^{m(k)}$, $p_k^- \in \mathbf{R}_+^{m(k)}$, $p^+ = \operatorname{vec}(\{p_k^+\})$, $p^- = \operatorname{vec}(\{p_k^-\})$, and $1_{m(k)} \in \mathbf{R}^{m(k)}$ is the vector of ones. Using this notation, the MAP estimation problem (5) is

$$\begin{array}{ll} \text{minimize} & S(x, p^+, p^-) \\ \text{w.r.t} & x \in \mathbf{R}^{Nn},\ p^+ \in \mathbf{R}_+^M,\ p^- \in \mathbf{R}_+^M \\ \text{subject to} & R^{-1/2}\bar{v}(x) = p^+ - p^-\,. \end{array} \quad (6)$$

Define $\tilde{S}$ as the approximate objective corresponding to the affine approximations $\{\tilde{g}_k\}$:

$$\begin{aligned} \tilde{g}_{k+1}(x_k;\, y_k) &= g_{k+1}(x_k) + g_{k+1}^{(1)}(x_k)(y_k - x_k)\,, \\ \tilde{S}(x;\, y, p^+, p^-) &= \sqrt{2}\,1_{m(k)}^T\,(p_k^+ + p_k^-) \\ &\quad + \frac{1}{2}\sum_{k=1}^{N}[y_k - \tilde{g}_k(x_{k-1}; y_{k-1})]Q_k^{-1}[y_k - \tilde{g}_k(x_{k-1}; y_{k-1})] \end{aligned}$$

and define $\tilde{v}(x;\ y)$ to be measurement residuals corresponding to the affine approximations $\{\tilde{h}_k\}$:

$$\begin{aligned}\tilde{h}_k(x_k;\ y_k) &= h_k(x_k) + h_k^{(1)}(x_k)(y_k - x_k)\\ \tilde{v}(x;\ y) &= \mathrm{vec}[\{z_k - \tilde{h}_k(x_k;\ y_k)\}]\ .\end{aligned}$$

Problem (6) is iteratively approximated by the quadratic programs (QP)

$$\begin{aligned}&\text{minimize} && \tilde{S}(x;\ y, p^+, p^-)\\ &\text{w.r.t} && y \in \mathbf{R}^{Nn}\ ,\ p^+ \in \mathbf{R}^M_+\ ,\ p^- \in \mathbf{R}^M_+\\ &\text{subject to} && R^{-1/2}\,\tilde{v}(x;\ y) = p^+ - p^-\ .\end{aligned} \tag{7}$$

Note that $\tilde{S}(x;\ \ y, p^+, p^-)$ is quadratic in y and affine in (p^+, p^-). We now rewrite (7) in standard QP format. Define $A_k(x) \in \mathbf{R}^{n\times n}$, $C_k(x) \in \mathbf{R}^{n\times n}$, and $C(x) \in \mathbf{R}^{nN\times nN}$ as follows:

$$\begin{aligned}A_k(x) &= -\,Q_k^{-1} g_k^{(1)}(x_{k-1})\ ,\\ C_k(x) &= Q_k^{-1} + g_{k+1}^{(1)}(x_k)^T Q_{k+1}^{-1} g_{k+1}^{(1)}(x_k)\ ,\\ C(x) &= \begin{bmatrix} C_1(x) & A_2^T(x) & 0 & \\ A_2(x) & C_2(x) & A_3^T(x) & 0 \\ 0 & \ddots & \ddots & \ddots \\ & 0 & A_N(x) & C_N(x)\end{bmatrix}\ .\end{aligned}$$

Here $C(x) = \partial_y \partial_y \tilde{S}(x;\ \ y, p^+, p^-)$. Next define the vector $a_k(x) \in \mathbf{R}^n$ by

$$\begin{aligned}a_k(x) &= \partial_{y(k)} \tilde{S}(x;\ y, p^+, p^-)|_{y=x}^T\\ &= Q_k^{-1}[x_k - g_k(x_{k-1})]\\ &\quad - g_{k+1}^{(1)}(x_k)^T Q_{k+1}^{-1}[x_{k+1} - g_{k+1}(x_k)]\end{aligned}$$

and define $a(x) \in \mathbf{R}^{nN}$ by $a(x) = \mathrm{vec}(\{a_k(x)\})$. Then

$$\begin{aligned}\tilde{S}(x;\ y, p^+, p^-) - \tilde{S}(x;\ x, p^+, p^-) &= \frac{1}{2} y^T C(x) y\\ &\quad + [a(x) - C(x)x]^T y + \frac{1}{2} x^T C(x) x - a(x)^T x\ .\end{aligned} \tag{8}$$

We also define the vectors $c(x) \in \mathbf{R}^{nN}$, $b(x) \in \mathbf{R}^{nN}$ and the matrix $B(x) \in \mathbf{R}^{nN\times nN}$ as follows:

$$\begin{aligned}c(x) &= a(x) - C(x)x\\ B(x) &= \mathrm{diag}\left[\left\{R_k^{-1/2}\, h_k^{(1)}(x_k)\right\}\right]\\ b(x) &= \mathrm{vec}\left(\left\{R_k^{-1/2}\,[z_k - h_k(x_k)]\right\}\right) - B(x)x\ .\end{aligned}$$

Dropping the terms in (8) that are constant with respect to y, subproblem (7) is equivalent to

$$\begin{aligned}&\text{minimize} && \frac{1}{2} y^T C(x) y + c(x)^T y + \sqrt{\mathbf{2}}^T (p^+ + p^-)\\ &\text{w.r.t.} && y \in \mathbf{R}^{nN}\ ,\ p^+ \in \mathbf{R}^M_+\ ,\ p^- \in \mathbf{R}^M_+\\ &\text{subject to} && b(x) + B(x) y = p^+ - p^-\end{aligned} \tag{9}$$

where (as above) $\sqrt{\mathbf{2}} \in \mathbf{R}^M$ is the vector with all its components equal to the square root of two. This is the Gauss–Newton subproblem that we use to solve the convex composite optimization problem (5).

Remark 3.1 (Huber Extension: The Map Subproblem): As noted in the introduction, it is possible to substitute the Huber distribution for the ℓ_1-Laplace distribution. In this remark we lay some of the groundwork required for this extension.

We are given a vector $\kappa \in \mathbf{R}^M$ of positive values. The Huber penalty function with threshold κ_i is defined as

$$\begin{aligned}\rho_i(t) &= \begin{cases} \kappa_i t - \frac{\kappa_i^2}{2} & \text{if } |t| \geq \kappa_i\\ \frac{t^2}{2} & \text{otherwise}\end{cases}\\ &= \max_s \left\{ st - \frac{s^2}{2} \,\middle|\, -\kappa_i \leq s \leq +\kappa_i \right\}\ .\end{aligned}$$

There are vectors $\zeta \in \mathbf{R}^M$ and $\xi \in \mathbf{R}^M$ of positive values such that for each i, $\exp[-\rho_i(\xi_i t)]/\zeta_i$ is a density for a probability distribution with mean zero and variance one on $t \in (-\infty, +\infty)$; see [1, eq. 5.4.8]. The vector ξ can be computed using the relation

$$\xi_i^2 = \frac{\sqrt{2\pi}\ \mathrm{erf}\left(\frac{\kappa_i}{\sqrt{2}}\right) + 4\exp\left(\frac{-\kappa_i^2}{2}\right)(\kappa_i^{-1} + \kappa_i^{-3})}{\sqrt{2\pi}\ \mathrm{erf}\left(\frac{\kappa_i}{\sqrt{2}}\right) + 2\exp\left(\frac{-\kappa_i^2}{2}\right)\kappa_i^{-1}}$$

where erf is the error function. The value of the vector ζ is not important for the discussion below. The multivariate Huber density corresponding to the measurement residuals $\bar{v}(x)$ is

$$\begin{aligned}\mathbf{p}[\bar{v}(x)] &= \prod_{i=1}^M \mathbf{p}\left(\left[R^{-1/2}\bar{v}(x)\right]_i\right)\\ &= \prod_{i=1}^M \frac{1}{\zeta_i}\exp\left[-\rho_i\left(\xi_i\left[R^{-1/2}\bar{v}(x)\right]_i\right)\right]\end{aligned}$$

where $\left[R^{-1/2}\bar{v}(x)\right]_i$ is the ith component of the vector $R^{-1/2}\bar{v}(x) \in \mathbf{R}^M$.

Dropping terms that do not affect the minimizer of the MAP objective (4), the estimation problem for the Huber case is equivalent to minimizing

$$\frac{1}{2}\bar{w}(x)^T Q^{-1}\bar{w}(x) + \sum_{i=1}^M \rho_i\left(\xi_i\left[R^{-1/2}\bar{v}(x)\right]_i\right)$$

with respect to $x \in \mathbf{R}^{Nn}$. Let $D(\xi) \in \mathbf{R}^{M\times M}$ be the diagonal matrix with ξ along the diagonal and define $\hat{b}(x) = D(\xi)b(x)$, $\hat{B}(x) = D(\xi)B(x)$, where $b(x)$ and $B(x)$ are as defined above. The linearized subproblem for the Huber case is

$$\text{minimize}\ \ \frac{1}{2} y^T C(x) y + c(x)^T y + \sum_{i=1}^M \rho_i([\hat{b}(x) + \hat{B}(x) y]_i)$$

where $C(x)$ and $c(x)$ are as defined above. In contrast to the ℓ_1-Laplace case (9), the problem above is not a convex quadratic program. However, it is an extended linear-quadratic convex program (see [40]), and our solution strategy preserves the underlying problem structure (see [1, Sec. 5.7.3]).

IV. Solving the QP Subproblem

The QP subproblem (9) can be solved using an interior point approach. Interior point methods apply a damped Newton's

method to a relaxation of the Karush–Kuhn–Tucker (KKT) conditions. The relaxed optimality conditions are themselves optimality conditions for an associated relaxed optimization problem given by

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}y^TCy + c^Ty + \sqrt{\mathbf{2}}^T(p^+ + p^-) \\ & - \tau\sum_{i=1}^{M}\log(p_i^+) - \tau\sum_{i=1}^{M}\log(p_i^-) \\ \text{w.r.t.} \quad & y \in \mathbf{R}^{nN},\ p^+ \in \mathbf{R}_+^M,\ p^- \in \mathbf{R}_+^M \\ \text{subject to} \quad & b + By - p^+ + p^- = 0\,. \end{aligned} \tag{10}$$

Here, and for the rest of this section, we drop the dependence of b, B, c, and C on x. The associated Lagrangian is

$$\begin{aligned} L_\tau(p^+, p^-, y, q) &= \frac{1}{2}y^TCy + c^Ty + \sqrt{\mathbf{2}}^T(p^+ + p^-) \\ &-\tau\sum_{i=1}^{M}\log(p_i^+) - \tau\sum_{i=1}^{M}\log(p_i^-) + q^T(b + By - p^+ + p^-) \end{aligned}$$

where q is a dual variable associated with the equality constraint. The corresponding KKT optimality conditions are

$$0 = \begin{bmatrix} \nabla_{p^+} \\ \nabla_{p^-} \\ \nabla_y \\ \nabla_q \end{bmatrix} L_\tau = \begin{bmatrix} \sqrt{\mathbf{2}} - q - \tau D(p^+)^{-1}\mathbf{1} \\ \sqrt{\mathbf{2}} + q - \tau D(p^-)^{-1}\mathbf{1} \\ Cy + c + B^Tq \\ b + By - p^+ + p^- \end{bmatrix} \tag{11}$$

where (as above) for any vector $v \in \mathbf{R}^M$, $D(v) \in \mathbf{R}^{M\times M}$ is the diagonal matrix with v along the diagonal. Define $s^+ \in \mathbf{R}_+^M$ and $s^- \in \mathbf{R}_+^M$ by the equations

$$D(p^+)D(s^+)\mathbf{1} - \tau\mathbf{1} = 0, \quad \text{and} \quad D(p^-)D(s^-)\mathbf{1} - \tau\mathbf{1} = 0.$$

Using the replacements $s^+ = \tau D(p^+)^{-1}\mathbf{1}$ and $s^- = \tau D(p^-)^{-1}\mathbf{1}$, and then subtracting the first two equations in the KKT conditions, we have

$$q = \frac{(s^- - s^+)}{2}\,. \tag{12}$$

Adding the first two equations in (11) and then replacing q in the third equation, the KKT optimality conditions are equivalent to the following function being equal to zero:

$$F_\tau \begin{pmatrix} p^+ \\ p^- \\ s^+ \\ s^- \\ y \end{pmatrix} = \begin{bmatrix} p^+ - p^- - b - By \\ D(p^-)D(s^-)\mathbf{1} - \tau\mathbf{1} \\ s^+ + s^- - 2\sqrt{\mathbf{2}} \\ D(p^+)D(s^+)\mathbf{1} - \tau\mathbf{1} \\ Cy + c + \frac{B^T(s^- - s^+)}{2} \end{bmatrix}. \tag{13}$$

The one-dimensional curve in the space of vectors (p^+, p^-, s^+, s^-, y) that satisfies the equation $F_\tau = 0$, with p^+, p^-, s^+ and s^- componentwise strictly positive, is called the *central path*. The idea behind the primal-dual interior point method is to apply a Newton based predictor-corrector method to follow the central path as $\tau \downarrow 0$.

The derivative matrix $F_\tau^{(1)}$ is given by

$$F_\tau^{(1)} \begin{pmatrix} p^+ \\ p^- \\ s^+ \\ s^- \\ y \end{pmatrix} = \begin{bmatrix} I & -I & 0 & 0 & -B \\ 0 & D(s^-) & 0 & D(p^-) & 0 \\ 0 & 0 & I & I & 0 \\ D(s^+) & 0 & D(p^+) & 0 & 0 \\ 0 & 0 & -\frac{B^T}{2} & \frac{B^T}{2} & C \end{bmatrix}.$$

This matrix is invertible at points p^+, p^-, s^+ and s^- that are componentwise strictly positive. At such points a step of Newton's method for locating roots of F_τ is computed by solving the linear system

$$F_\tau^{(1)} \begin{pmatrix} p^+ \\ p^- \\ s^+ \\ s^- \\ y \end{pmatrix} \begin{bmatrix} \Delta p^+ \\ \Delta p^- \\ \Delta s^+ \\ \Delta s^- \\ \Delta y \end{bmatrix} = -F_\tau \begin{pmatrix} p^+ \\ p^- \\ s^+ \\ s^- \\ y \end{pmatrix} \tag{14}$$

for Δp^+, Δp^-, Δs^+, Δs^-, and Δy. Elementary block row operations can be used to reduce (14) to an upper block triangular system. Doing so proves the following lemma:

Lemma 4.1: Given the inputs p^+, p^-, s^+, s^-, y, B, b, C, and c, the following algorithm solves (14) for Δp^+, Δp^-, Δs^+, Δs^-, and Δy:

$$\begin{aligned} \bar{d} &= \frac{\tau\mathbf{1}}{s^+} - \frac{\tau\mathbf{1}}{s^-} - b - By + p^+ \\ \bar{e} &= B^T(\sqrt{\mathbf{2}} - s^-) - Cy - c \\ \bar{f} &= \bar{d} - D(s^+)^{-1}D(p^+)(2\sqrt{\mathbf{2}} - s^-) \\ T &= D(s^+)^{-1}D(p^+) + D(s^-)^{-1}D(p^-) \\ \Delta y &= [C + B^TT^{-1}B]^{-1}(\bar{e} + B^TT^{-1}\bar{f}) \\ \Delta s^- &= T^{-1}B\Delta y - T^{-1}\bar{f} \\ \Delta s^+ &= -\Delta s^- + 2\sqrt{\mathbf{2}} - s^+ - s^- \\ \Delta p^- &= D(s^-)^{-1}[\tau\mathbf{1} - D(p^-)\Delta s^-] - p^- \\ \Delta p^+ &= \Delta p^- + B\Delta y + b + By - p^+ + p^-. \end{aligned}$$

The matrix T in the lemma above is diagonal with all the diagonal elements positive. The matrix $C + B^TT^{-1}B \in \mathbf{R}^{Nn\times Nn}$ is positive definite (because C is) and block tridiagonal with block size n. We can multiply the inverse of this matrix times a vector in $O(n^3 \times N)$[5, Alg. 4]; thus, the work required grows linearly with the number of time steps.

The interior point algorithm proceeds by applying a damped Newton's method to the system of nonlinear equations $F_\tau(p^+, p^-, s^-, s^+, y) = 0$, where the damping is employed to maintain the strict positivity of the iterates $(p^+, p^-, s^-, s^+, y)^\nu$. The *central path* parameter τ is simultaneously reduced to zero so that in the limit the optimality conditions for (9) are satisfied. We do not provide details for our implementation of the interior point algorithm since they are easily found elsewhere, but we do briefly discuss the duality theory for (9) since it impacts both the termination criteria for the interior point algorithm and our choice of search direction in solving (5).

The dual to the quadratic program (9) is the quadratic program

$$\begin{aligned} &\text{maximize} \quad -\frac{1}{2}(c+B^Tq)^TC^{-1}(c+B^Tq)+b^Tq \\ &\text{w.r.t.} \quad q\in\mathbf{R}^M \\ &\text{subject to} \quad -\sqrt{\mathbf{2}}\le q\le\sqrt{\mathbf{2}} \end{aligned} \tag{15}$$

(where again the dependence of b, B, c, and C on x has been dropped). Given a primal feasible point (y,p^+,p^-) for problem (9) and its associated primal objective value

$$V_P(y,p^+,p^-)=\frac{1}{2}y^TC(x)y+c(x)^Ty+\sqrt{\mathbf{2}}^T(p^++p^-)$$

as well as a dual feasible point q for problem (15) and its associated dual objective value

$$V_D(q)=-\frac{1}{2}(c+B^Tq)^TC^{-1}(c+B^Tq)+b^Tq$$

we say that $V_P(y,p^+,p^-)-V_D(q)$ is the duality gap associated with the primal-dual pair $[(y,p^+,p^-),q]$. This value is necessarily non-negative, taking the value zero if and only if (y,p^+,p^-) is primal optimal and q is dual optimal.

Lemma 4.2: Suppose y,p^+,p^-,s^+,s^- satisfy the affine equations

$$\begin{aligned} 0&=p^+-p^--b-By\,, \qquad 0=s^++s^--2\sqrt{\mathbf{2}} \\ 0&=Cy+c+\frac{B^T(s^--s^+)}{2} \end{aligned} \tag{16}$$

with p^+, p^-, s^+, s^- componentwise strictly positive. If we choose q as in (12), then $[(y,p^+,p^-),q]$ is a primal-dual feasible pair and $(s^+)^Tp^++(s^-)^Tp^-$ is the associated duality gap.

Proof: If (16) and (12) hold then $c+B^Tq=-Cy$, $V_D(q)=-(1/2)y^TCy+b^Tq$ and equals

$$\begin{aligned} &-\frac{1}{2}y^TCy+\frac{(p^+-p^--By)^T(s^--s^+)}{2} \\ &=-\frac{1}{2}y^TCy+y^T(Cy+c)+\frac{(p^+-p^-)^T(s^--s^+)}{2} \\ &=\frac{1}{2}y^TCy+c^Ty+\frac{(s^++s^-)^T(p^++p^-)}{2} \\ &\quad-(s^+)^Tp^+-(s^-)^Tp^- \\ &=\frac{1}{2}y^TCy+c^Ty+\sqrt{\mathbf{2}}^T(p^++p^-) \\ &\quad-\left[(s^+)^Tp^++(s^-)^Tp^-\right] \\ &=V_P(y,p^+,p^-)-\left[(s^+)^Tp^++(s^-)^Tp^-\right]\,. \end{aligned}$$

■

Remark 4.1 (Huber Extension: Path-Following): When the ℓ_1-Laplace density is replaced by the Huber density, the relaxed version of the KKT optimality conditions is $F_\tau=0$ where

$$F_\tau\begin{pmatrix}p^+\\p^-\\s^-\\s^+\\y\end{pmatrix}=\begin{bmatrix}p^+-p^--\hat{b}-\hat{B}y+s^+-s^-\\D(p^-)D(s^+)\mathbf{1}-\tau\mathbf{1}\\s^++s^--\kappa\\D(p^+)D(s^-)\mathbf{1}-\tau\mathbf{1}\\Cy+c+\hat{B}^T(s^+-s^-)\end{bmatrix}\,.$$

Following the procedure outlined above for the ℓ_1 case, one can show that a Newton-based predictor-corrector path-following method that follows the central path for this function also preserves the underlying block tridiagonal structure of the original problem.

V. Solving the Map Estimation Problem

We employ a standard generalized Gauss–Newton algorithm for the convex composite structure (5) with a simple backtracking line search. The objective function in (5) has the form $K=\rho\circ F$, where the convex function ρ and the smooth function F are given by

$$\rho\begin{pmatrix}v\\w\end{pmatrix}=\sqrt{2}\left\|R^{-1/2}v\right\|_1+\frac{1}{2}w^TQ^{-1}w \tag{17}$$

$$F(x)=\begin{pmatrix}\bar{v}(x)\\\bar{w}(x)\end{pmatrix}\,. \tag{18}$$

The first-order necessary condition for optimality in the convex composite problem minimize $K(x)$ is $0\in\partial K(x)=\partial\rho[F(x)]F^{(1)}(x)$, where $\partial K(x)$ is the generalized subdifferential of K at x[40] and $\partial\rho[F(x)]$ is the convex subdifferential of ρ at $F(x)$[39]. Elementary convex analysis gives us the equivalence

$$0\in\partial K(x)\quad\Leftrightarrow\quad K(x)=\inf_y\rho[\,F(x)+F^{(1)}(x)(y-x)\,]\,.$$

This motivates our definition of the difference function $\Delta(x;\ d)=\rho[\,F(x)+F^{(1)}(x)d\,]-K(x)$, and its minimum with respect to direction $\Delta^*(x)=\inf_d\Delta(x;\ d)$. Hence, $\Delta^*(x)=0$ if and only if $0\in\partial K(x)$. The directional derivative of K is given by [10]

$$K^{(1)}(x;\ d)=\lim_{t\downarrow0}t^{-1}\Delta(x;\ td)=\inf_{t>0}t^{-1}\Delta(x;\ td)\,.$$

Given $\eta\in(0,1)$, we define a set of descent directions at x by $D(x,\eta)=\{d\mid\Delta(x;\ d)\le\eta\Delta^*(x)\}$ (not to be confused with $D(x)$, the diagonal matrix with x on the diagonal). Note that if there is a $d\in D(x,\eta)$ such that $\Delta(x;\ d)\ge-\eta\varepsilon$, then $\Delta^*(x)\ge-\varepsilon$. These ideas motivate the following application of the Gauss–Newton algorithm in [9].

Algorithm 5.1: The Iterated ℓ_1-Laplace Kalman Smoother:

1) (Initialize) We are given $x^0\in\mathbf{R}^{Nn}$ an initial estimate of state sequence, $\varepsilon\ge0$ an overall termination criteria, $\eta\in(0,1)$ a termination criteria for subproblem, $\beta\in(0,1)$ a line search rejection criteria, $\gamma\in(0,1)$ a line search step size factor. Set the iteration counter $\nu=0$.
2) (Gauss–Newton Step) Find y^ν an approximate solution to the QP subproblem (9) at $x=x^\nu$. To be specific, $d^\nu=y^\nu-x^\nu$ is in the set $D(x^\nu,\eta)$. Set $\Delta_\nu=\Delta(x^\nu;\ d^\nu)$ and *Terminate* if $\Delta_\nu\ge-\varepsilon$.
3) (Line Search) Set t_ν to the maximum element of
$$\left\{\gamma^i\;\middle|\;\begin{array}{c}i\in\{0,1,2,\ldots\}\text{ and}\\ \rho[F(x^\nu+\gamma^id^\nu)]\le\rho[F(x^\nu)]+\beta\gamma^i\Delta_\nu\end{array}\right\}\,.$$
4) (Iterate) Set $x^{\nu+1}=x^\nu+t_\nu d^\nu$ and goto Step 2.

Remark 5.1 (Existence of Solutions to (9)): Due the coercivity of ρ(17) there always exists a vector $d^*\in\mathbf{R}^{Nn}$ such that

$\Delta(x;\ d^*) = \Delta^*(x)$. Indeed, the coercivity of ρ implies that the set $\{u \mid \rho(u) \leq K(x)\}$ is compact, and so the set

$$U(x) = \{u \mid \rho(u) \leq K(x)\} \cap \left(F(x) + \text{range}\left[F^{(1)}(x)\right]\right)$$

is also compact since the range of $F^{(1)}(x)$ is a closed subspace. It follows that $\Delta^*(x) = \inf\{\rho(u) - K(x) \mid u \in U(x)\}$, and there is a $u^* \in U(x)$ such that $\Delta^*(x) = \rho(u^*)$ (because $U(x)$ is compact). Let $d^* \in \mathbf{R}^{Nn}$ be such that $u^* = F(x) + F^{(1)}(x)d^*$. It follows that $\Delta^*(x) = \rho(u^*) = \Delta(x;\ d^*)$.

Remark 5.2 (Implementation of Step 2): Step 2 of the algorithm specifies that the search direction d^ν satisfies

$$\Delta(x^\nu; d^\nu) \leq \eta\Delta^*(x^\nu)\ . \tag{19}$$

In this remark we describe how this can be achieved using the primal-dual interior point algorithm of the previous section. We begin by translating condition (19) into the notation of Lemma 4.2. For a fixed y in problem (9), the optimal choices for p^+ and p^- are

$$\tilde{p}^+(x,y) = [b(x) + B(x)y]_+,\ \tilde{p}^-(x,y) = -[b(x) + B(x)y]_-$$

where for any vector $u \in \mathbf{R}^M$ the vectors u_+ and u_- are defined componentwise by $(u_+)_i = \max\{0, u_i\}$, and $(u_-)_i = \min\{0, u_i\}$. We define the corresponding optimal value function

$$\tilde{V}_P(x,y) = V_P[x, y, \tilde{p}^+(x,y), \tilde{p}^-(x,y)]\ .$$

The relationships between problem (7), (8), and problem (9), show that there is a function $\theta(x)$, that does not depend on y, such that

$$\begin{aligned}\tilde{V}_P(x,y) + \theta(x) &= \tilde{S}[x;\ y, \tilde{p}^+(x,y), \tilde{p}^-(x,y)]\\ &= \rho[F(x) + F^{(1)}(x)(y-x)]\\ \tilde{V}_P(x,x) + \theta(x) &= \rho[F(x)] = K(x)\ .\end{aligned}$$

Using $d^\nu = y^\nu - x^\nu$, inequality (19) becomes

$$\tilde{V}_P(x^\nu, y^\nu) - \tilde{V}_P(x^\nu, x^\nu) \leq \eta[V_P^*(x^\nu) - \tilde{V}_P(x^\nu, x^\nu)]$$

where $V_P^*(x^\nu)$ denotes the optimal value of the QP subproblem (9). Adding

$$\tilde{V}_P(x^\nu, x^\nu) - \eta\tilde{V}_P^*(x^\nu) - (1-\eta)\tilde{V}_P(x^\nu, y^\nu)$$

to both sides of the previous inequality we obtain

$$\eta[\tilde{V}_P(x^\nu, y^\nu) - V_P^*(x^\nu)] \leq (1-\eta)[\tilde{V}_P(x^\nu, x^\nu) - \tilde{V}_P(x^\nu, y^\nu)].$$

We observe that, from elementary duality theory, $V_D(x^\nu, q) \leq V_P^*(x^\nu)$ for every point q that is feasible for the dual problem (15). We also observe that, if (y^ν, p^+, p^-) is primal feasible

$$\tilde{V}_P(x^\nu, y^\nu) \leq V_P(x^\nu, y^\nu, p^+, p^-)\ .$$

Combining these two observations with the preceding inequality, we find that condition (19) is satisfied whenever

$$\begin{aligned}&\eta[V_P(x^\nu, y^\nu, p^+, p^-) - V_D(x^\nu, q)]\\ &\quad\leq (1-\eta)[\tilde{V}_P(x^\nu, x^\nu) - V_P(x^\nu, y^\nu, p^+, p^-)]\end{aligned} \tag{20}$$

where q is any point that is dual feasible and p^+, p^- are such that (y^ν, p^+, p^-) is primal feasible. The left-hand side of (20) is η times the duality gap associated with the current primal-dual feasible pair $[(y^\nu, p^+, p^-), q]$. This gap is driven to zero by the interior point algorithm, while the right-hand side is driven to the value $-(1-\eta)\Delta^*(x^\nu)$. If x^ν is not a stationary point of problem (5), then $0 < -\Delta^*(x^\nu)$. In this case, inequality (20) must be satisfied by the iterates of the interior point algorithm after finitely many iterations (and hence condition (19) is also satisfied).

Lemma 4.2 can be used to simplify condition (20). This lemma states that if $(p^+, p^-, s^+, s^-, y, q)$ satisfies the affine equations (16) and (12) with p^+, p^-, s^+, and s^- all strictly positive, then

$$V_P(x^\nu, y, p^+, p^-) - V_D(x^\nu, q) = (s^+)^T p^+ + (s^-)^T p^-\ .$$

Moreover, if the interior point algorithm is initialized at a point satisfying the affine equations (16) and (12), then by (14) all subsequent iterates must satisfy these equations as well. In this case,

$$||F_0(p^+, p^-, s^+, s^-, y)||_1 = (s^+)^T p^+ + (s^-)^T p^- \tag{21}$$

for all interior point iterates, and the criteria (20) becomes

$$\begin{aligned}&\eta[(s^+)^T p^+ + (s^-)^T p^-]\\ &\quad\leq (1-\eta)[\tilde{V}_P(x^\nu, x^\nu) - V_P(x^\nu, y^\nu, p^+, p^-)]\ .\end{aligned} \tag{22}$$

In our implementation, given an initial value for the homotopy parameter τ, we initialize the interior point algorithm at the point

$$\begin{aligned}y &= -C(x^\nu)^{-1}c(x^\nu)\ , \qquad s^+ = s^- = \sqrt{\mathbf{2}}\\ p^+ &= [b(x^\nu) + B(x^\nu)y]_+ + \frac{\tau}{\sqrt{2}}\mathbf{1}\\ p^- &= -[b(x^\nu) + B(x^\nu)y]_- + \frac{\tau}{\sqrt{2}}\mathbf{1}\ .\end{aligned}$$

We initialize $q = 0$ and use (22) as the stopping criteria. This initial point satisfies the affine equations (16) and (12). Thus, (19) is satisfied at termination, and, by (21), the accuracy of the subproblem solution automatically adjusts with the difference of the objective values. That is, as $\Delta^*(x^\nu)$ increases to zero, the accuracy required in the QP subproblem solution automatically increases.

Theorem 5.1: Let ρ and F be as in (17) and (18), respectively, and further assume that $x^0 \in \mathbf{R}^{Nn}$ is such that $F^{(1)}$ is uniformly continuous on the set $\overline{\text{co}}\left(\{x \mid K(x) \leq K(x^0)\}\right)$. If $\{x^\nu\}$ is a sequence generated by the Gauss–Newton Algorithm above with initial point x^0 and $\varepsilon = 0$, then one of the following must occur:

(i) The algorithm terminates finitely at a point x^ν with $0 \in \partial K(x^\nu)$.
(ii) $\lim_{\nu\in I} \Delta_\nu = 0$ for every subsequence I for which the set $\{d^\nu \mid \nu \in I\}$ is bounded.
(iii) The sequence $||d^\nu||$ diverges to $+\infty$.

Moreover, if $\bar{x}$ is any cluster point of a subsequence $I \subset \mathbf{Z}_+$ such that the subsequence $\{d^\nu\}_I$ is bounded, then $0 \in \partial K(\bar{x})$.

Proof: The first three assertions are a restatement of [9, Theorem 2.4] in our context, where the sets D_ν in [9, Theorem 2.4] are given by $D_\nu = D(x^\nu, \eta)$. The requirement that ρ is Lipschitz continuous on the set $\{u \,|\, \rho(u) \le K(x^0)\}$ is an immediate consequence of the fact that ρ is coercive, so this set is compact. This completes the proof of (i), (ii), and (iii). We now prove the final assertion of this theorem.

Suppose $\bar{x}$ is a cluster point of a sequence $I \subset \mathbf{Z}_+$ for which $\{d^\nu\}_I$ is bounded. We can assume that there is a subsequence $J \subset I$ and a vector $\bar{d} \in \mathbf{R}^{Nn}$ such that $(x^\nu, d^\nu) \to_J (\bar{x},\ \bar{d})$. Fix any other point $\hat{d} \in \mathbf{R}^{Nn}$. By construction, we know that

$$\begin{aligned}\Delta_\nu &= \rho[\, F(x^\nu) + F^{(1)}(x^\nu)d^\nu\,] - \rho[F(x^\nu)] \le \eta\Delta^*(x^\nu)\\ &\le \eta\{\rho[\, F(x^\nu) + F^{(1)}(x^\nu)\hat{d}\,] - \rho[F(x^\nu)]\}\ .\end{aligned}$$

Taking the limit over J gives

$$\begin{aligned}0 &= \rho[\, F(\bar{x}) + F^{(1)}(\bar{x})\bar{d}\,] - \rho[F(\bar{x})]\\ &\le \eta\{\rho[\, F(\bar{x}) + F^{(1)}(\bar{x})\hat{d}\,] - \rho[F(\bar{x})]\}\ .\end{aligned}$$

But $\hat{d}$ was an arbitrary point in $\mathbf{R}^{Nn}$ and so $\Delta^*(\bar{x}) = 0$. ∎

A stronger convergence result is possible under stronger assumptions on F and $F^{(1)}$.

Corollary 5.1: Let ρ and F be as in (17) and (18), respectively, and recall that $M = \sum_{k=1}^N m(k)$. Fix $x^0 \in \mathbf{R}^{Nn}$, define $\Lambda = \{u \in \mathbf{R}^M \times \mathbf{R}^{Nn} \,|\, \rho(u) \le K(x^0)\}$, suppose that $F^{-1}(\Lambda) = \{x \,|\, F(x) \in \Lambda\}$ is bounded, and

$$\text{kernel}[F^{(1)}(x)] = \{0\} \quad \forall\, x \in F^{-1}(\Lambda)\ . \tag{23}$$

If $\{x^\nu\}$ is a sequence generated by the Gauss–Newton algorithm above with initial point x^0 and $\varepsilon = 0$, then $\{x^\nu\}$ and $\{d^\nu\}$ are bounded and either the algorithm terminates finitely at a point x^ν with $0 \in \partial K(x^\nu)$, or $\Delta_\nu \to 0$ as $\nu \to \infty$. Moreover, every cluster point $\bar{x}$ of the sequence $\{x^\nu\}$ satisfies $0 \in \partial K(\bar{x})$.

Remark 5.3: The condition (23) is guaranteed to hold if each of the matrices $g_k^{(1)}(x_k),\ k = 1, \ldots, N$ is nonsingular for all $x \in F^{-1}(\Lambda)$.

Proof: First note that $F^{-1}(\Lambda)$ is closed since F is continuous and Λ is compact, therefore, $F^{-1}(\Lambda)$ is compact. Hence, $\overline{\text{co}}\,[F^{-1}(\Lambda)]$ is also compact. Therefore, $F^{(1)}$ is uniformly continuous on $\overline{\text{co}}\,[F^{-1}(\Lambda)]$ which implies that the hypotheses of Theorem 5.1 are satisfied, and so one of (i)-(iii) must hold. If (i) holds we are done, so we will assume that the sequence $\{x^\nu\}$ is infinite. Since $\{x^\nu\} \subset F^{-1}(\Lambda)$, this sequence is bounded. We now show that the sequence $\{d^\nu\}$ of search directions is also bounded.

We claim that there exists $\lambda > 0$ such that $\lambda\|d\| \le \|F^{(1)}(x)d\|$ for all $d \in \mathbf{R}^{Nn}$ and $x \in F^{-1}(\Lambda)$. Indeed, if this were not the case, then there would exist sequences $\{y^i\} \subset F^{-1}(\Lambda)$ and $\{d^i\} \subset \mathbf{R}^{Nn}$ such that $d^i \ne 0$ and $\|d^i\|/i > \|F^{(1)}(y^i)d^i\| \quad \forall\, i = 1, 2, \ldots$. The set $F^{-1}(\Lambda)$ is compact; hence, there exist a subsequence $J \subset \mathbf{Z}_+$, vector $\bar{x} \in F^{-1}(\Lambda)$, and vector $\bar{d} \in \mathbf{R}^{Nn}$ with $\|\bar{d}\| = 1$, such that $x^\nu \to_J \bar{x}$ and $d^i/\|d^i\| \to_J \bar{d}$. It follows from the inequality above that $1/i \ge \|\ F^{(1)}(x^i)d^i/\|d^i\|\ \|$, and taking the limit with respect to the subsequence J we obtain $0 \ge \|F^{(1)}(\bar{x})\bar{d}\|$. Thus, $\bar{d}$ is in the kernel of $F^{(1)}(\bar{x})$ and $\bar{d} \ne 0$. This contradicts the assumption that the kernel of $F^{(1)}(x)$ is the singleton $\{0\}$ and thereby proves the claim.

Since Λ is compact, there is a $\alpha > 0$ such that $\|u\| \le \alpha$ for all $u \in \Lambda$ and for $\nu = 1, 2, \ldots$,

$$\begin{aligned}\lambda\|d^\nu\| &\le \left\| F^{(1)}(x^\nu)d^\nu \right\|\\ &\le \left\| F(x^\nu) + F^{(1)}(x^\nu)d^\nu \right\| + \|F(x^\nu)\| \le 2\alpha\end{aligned}$$

since $\{F(x^\nu),\ F(x^\nu) + F^{(1)}(x^\nu)d^\nu\} \subset \Lambda$ by construction. Hence, the sequence $\{d^\nu\}$ of search directions is bounded. Hence Theorem 5.1 tells us that $\Delta_\nu \to 0$ as $\nu \to \infty$.

The final statement of the corollary follows immediately from the final statement of Theorem 5.1. ∎

VI. Method Comparison Using Simulated Data

We implemented the Iterated ℓ_1-Laplace Kalman Smoother (ILS) in Algorithm 5.1. In this section, we compare its estimates with the Iterated Gaussian Kalman Smoother (IGS) estimates computed by [7] (without constraints), with the Gaussian Kalman Filter (GKF) estimates computed by [36], and with the Robust Kalman Filter (RKF) estimates computed by [41].

The simulated numerical experiments have four components: the state sequence simulation, the measurement sequence simulation, the specification of the mathematical model in (3), and the method for reporting the experimental results. In applications the observed phenomenon typically does not obey the assumptions posited by its mathematical model [8]. That is, modeling error is always present. In (3), the role of the model functions $g_k(x_{k-1})$ and $h_k(x_k)$ is to capture our knowledge and beliefs while the random vectors w_k and v_k attempt to capture our uncertainty and ignorance. Both experiments described below exhibit some modeling error. Of particular importance is the modeling error present in the measurements which we refer to as outlier data. The goal is to show that the proposed estimation procedure is robust with respect to this error. The four components of our numerical experiments are detailed below.

1) State Sequence Simulation: We call the simulated state sequence $\{x_k\}$ "ground truth." The first experiment is linear and its ground truth is obtained by sampling from a smooth nonlinear trajectory in the plane. Although the resulting state sequence is deterministic, it does not satisfy the transition equation $x_k = g_k(x_{k-1})$ which models, up to first order, our belief that the trajectory is smooth. The second experiment is nonlinear and the ground truth is obtained by specifying x_0 and generating the sequence $\{x_k\}$ from the transition equation $x_k = g_k(x_{k-1}) + w_k$, where $\{w_k\}$ is independent and identically distributed Gaussian. Thus, the second experiment has a stochastic state sequence and this sequence satisfies the model assumptions in (3).

2) Measurement Sequence Simulation: Since we study robustness with respect to the measurement sequence, the measurements should not satisfy the assumptions in (3). That is, the smoother has an imperfect model for the sequence $\{v_k\}$. We simulate from a mixture of two normal distributions: the nominal distribution and the outlier distribution. We vary the proportions of the mixture as well as the variance of the outliers.

3) Mathematical Model Specification: We specify the functions $g_k(x_{k-1})$, $h_k(x_k)$, and the variance matrices Q_k, R_k in (3). This data is known to the smoother.

TABLE I
MEDIAN MSE AND 95% CONFIDENCE INTERVALS FOR THE DIFFERENT ESTIMATION METHODS

p	ϕ	GKF	RKF	IGS	ILS
0	–	.34 (.24, .47)	.42 (.15, 1.1)	.04(.02, .1)	.04(.01, .1)
.1	1	.41(.26, .60)	.48 (.15, 1.1)	.06(.02, .12)	.04(.02, .10)
.1	4	.59(.32, 1.1)	.56 (.18, 1.5)	.09(.04, .29)	.05(.02, .12)
.1	10	1.0(.42, 2.3)	.58 (.19, 1.7)	.17(.05, .55)	.05(.02, .13)
.1	100	6.8(1.7, 17.9)	.55 (.18, 2.0)	1.3(.30, 5.0)	.05(.02, .14)

TABLE II
MEDIAN MSE OVER 1000 RUNS AND CONFIDENCE INTERVALS CONTAINING 95% OF MSE RESULTS

$\mathbf{p}$	ϕ	IGS	ILS
0	–	0.07 (0.06, 0.08)	0.07 (0.06, 0.09)
.1	10	0.07 (0.06, 0.10)	0.07 (0.06, 0.09)
.2	10	0.08 (0.06, 0.11)	0.08 (0.06, 0.11)
.3	10	0.08 (0.06, 0.11)	0.08 (0.06, 0.11)
.1	100	0.10 (0.07, 0.14)	0.07 (0.06, 0.10)
.2	100	0.12 (0.07, 0.40)	0.08 (0.06, 0.11)
.3	100	0.13 (0.09, 0.64)	0.08 (0.07, 0.10)
.1	1000	0.17 (0.11, 1.50)	0.08 (0.06, 0.11)
.2	1000	0.21 (0.14, 2.03)	0.08 (0.06, 0.11)
.3	1000	0.25 (0.17, 2.66)	0.09 (0.07, 0.12)

4) Reporting Results: The experimental results are condensed and reported as the median of the mean square errors (MSE) with respect to 1000 replications of each experiment. We also provide 95% confidence intervals for the MSE, i.e., intervals that contain 95% of the MSE results. The results in Table I are for the linear problem and compare the quality of estimates between filters and smoothers as well as between robust and least squares fitting. The results in Table II are for a nonlinear problem and hence only the iterated smoothers are compared.

A. Unknown Linear Deterministic Process

For this simulation, the "ground truth" state vector satisfies the following deterministic linear differential equation:

$$X(0) = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \ \dot{X}(t) = \begin{pmatrix} -X_2(t) \\ X_1(t) \end{pmatrix}$$
$$\text{i.e.,} \qquad X(t) = \begin{pmatrix} -\cos(t) \\ -\sin(t) \end{pmatrix}.$$

For $k = 0, \ldots, N$, let $t_k = k\Delta t$ and $x_k = X(t_k)$. We simulate a lack of knowledge of the ground truth using the following stochastic process model for the mean of x_k given x_{k-1}

$$g_k(x_{k-1}) = \begin{bmatrix} 1 & 0 \\ \Delta t & 1 \end{bmatrix} x_{k-1}.$$

In other words, the first component is modeled as a random walk and the second as the integral of the first plus random noise. The process model for the variance of x_k given x_{k-1} is

$$Q_k = \begin{bmatrix} \Delta t & \frac{\Delta t^2}{2} \\ \frac{\Delta t^2}{2} & \frac{\Delta t^3}{3} \end{bmatrix}.$$

This model is discussed in more detail in [6, Sec. 8].

Two full periods of $X(t)$ were generated with $N = 100$ and $\Delta t = 4\pi/N$; i.e., discrete time points equally spaced over the interval $[0, 4\pi]$. For $k = 1, \ldots, N$ the measurements z_k were simulated by $z_k = X_2(t_k) + v_k$. The measurement noise v_k was generated as a mixture of two normals with p denoting the fraction of outlier contamination; i.e.,

$$v_k \sim (1-p)\mathbf{N}(0, 0.25) + p\mathbf{N}(0, \phi). \tag{24}$$

This was done for $p \in \{0,\ 0.1\}$ and $\phi \in \{1, 4, 10, 100\}$. The model for the mean of z_k given x_k is $h_k(x_k) = (0,1)x_k = x_{2,k}$, where $x_{2,k}$ denotes the second component of x_k. The model for the variance of z_k given x_k is $R_k = 0.25$. This simulates a lack of knowledge of the distribution for the outliers; i.e., $p\mathbf{N}(0, \phi)$. Note that we are recovering estimates for the smooth function $-\sin(t)$ and its derivative $-\cos(t)$ using noisy measurements (with outliers) of the function values.

We simulated 1000 realizations of the sequence $\{z_k\}$ keeping the ground truth fixed, and for each realization, and each estimation method, we computed the corresponding state sequence estimate $\{\hat{x}_k\}$. The MSE corresponding to such an estimate is defined by

$$\text{MSE} = \frac{1}{N}\sum_{k=1}^{N}[x_{1,k} - \hat{x}_{1,k}]^2 + [x_{2,k} - \hat{x}_{2,k}]^2 \tag{25}$$

where $x_k = X(t_k)$. In Table I, the Gaussian Kalman Filter is denoted by (GKF), the Robust Kalman Filter (RKF), the Iterated Gaussian Smoother (IGS), and the Iterated ℓ_1-Laplace Smoother (ILS). (The RKF used its default efficiency setting 0.9 [41].) For each of these estimation techniques, each value of p, and each value of ϕ, the corresponding table entry is the median MSE followed by the centralized 95% confidence interval for the MSE. For this problem, the model functions $\{g_k(x_{k-1})\}$ and $\{h_k(x_k)\}$ are linear so the iterated smoothers IGS and ILS only require one iteration to compute an estimate sequence $\{\hat{x}_k\}$.

Note the ℓ_1-Laplace smoother performs nearly as well as the Gaussian smoother at the nominal conditions ($p = 0$). The ℓ_1-Laplace smoother performs better and more consistently in cases with data contamination ($p \geq .1$ and $\phi \geq 1$). We also see that the robust filter performs better than the standard filter in cases with data contamination. It is also apparent that the smoothers perform better than the filters.

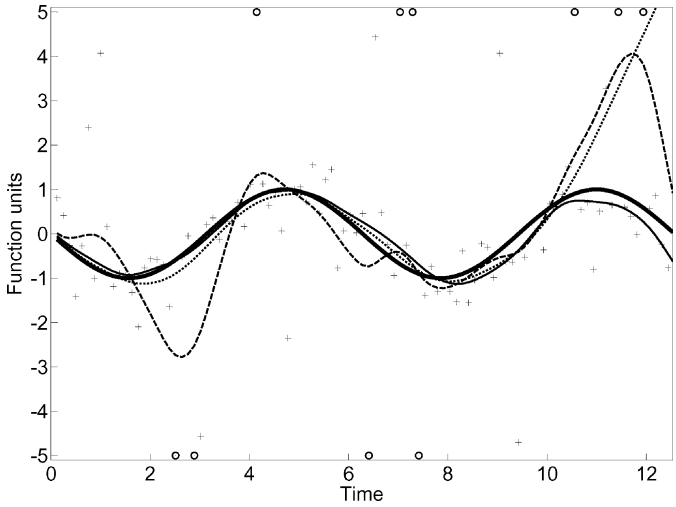

Fig. 2. Simulation: measurements (+), outliers (o) (absolute residuals more than three standard deviations), true function (thick line), ℓ_1-Laplace estimate (thin line), Gaussian estimate (dashed line), Gaussian outlier removal estimate (dotted line).

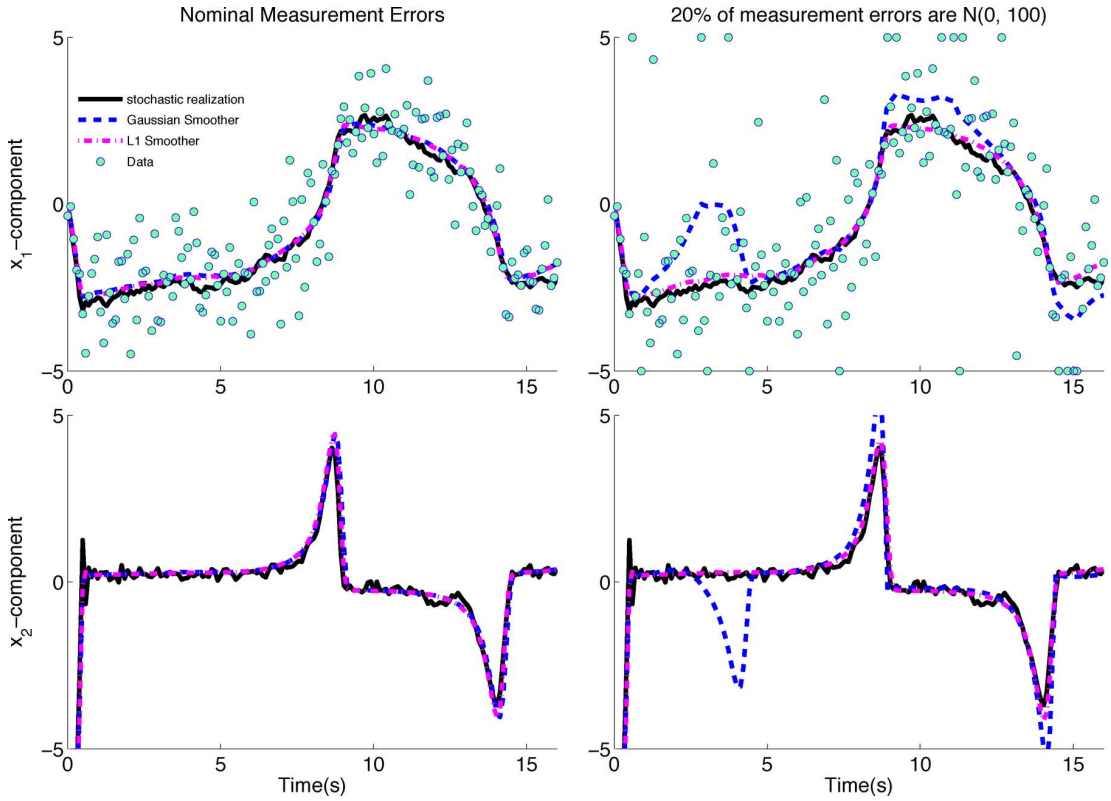

Fig. 3. The left two panels show estimation of x_1 (top) and x_2 (bottom) with errors from the nominal model. The stochastic realization is represented by a thick black line; the Gaussian smoother is the blue dashed line, and the ℓ_1-smoother is the magenta dash-dotted line. Right two panels show the same stochastic realization but with measurement errors now from $(p, \phi) = (.2, 100)$. Outliers appear on the top and bottom boundary in the top right panel.

Outlier detection and removal followed by refitting is a simple approach to robust estimation and can be applied to the smoothing problem. An inherent weakness of this approach is that the outlier detection is done using an initial fit which assumes outliers are not present. This can lead to good data being classified as outliers and result in over fitting the remaining data. An example of this is illustrated in Fig. 2 which plots the estimation results for a realization of $\{z_k\}$ where $p = 0.1$ and $\phi = 100$. Outlier removal also makes critical review of the model more difficult. A robust smoothing method with a consistent model, such as the ℓ_1-Laplace smoother, does not suffer from these difficulties.

B. Stochastic Nonlinear Process

The Van der Pol oscillator is a popular nonlinear process for comparing Kalman filters; e.g., [21] and [28, Sec. 4.1]. The corresponding nonlinear differential equation is

$$\dot{X}_1(t) = X_2(t) \text{ and } \dot{X}_2(t) = \mu[1 - X_1(t)^2]X_2(t) - X_1(t) .$$

Given $X(t_{k-1}) = x_{k-1}$ the Euler approximation for $X(t_{k-1} + \Delta t)$ is

$$g_k(x_{k-1}) = \begin{pmatrix} x_{1,k-1} + x_{2,k-1}\Delta t \\ x_{2,k-1} + \{\mu[1 - x_{1,k}^2]x_{2,k} - x_{1,k}\}\Delta t \end{pmatrix} .$$

For this simulation, the "ground truth" is obtained from a stochastic Euler approximation of the Van der Pol oscillator. To be specific, with $\mu = 2$, $N = 164$, and $\Delta t = 16/N$, the ground truth state vector x_k at time $t_k = k\Delta t$ is given by $x_0 = (0, -0.5)^T$ and for $k = 1, \ldots, N$:

$$x_k = g_k(x_{k-1}) + w_k \tag{26}$$

where $\{w_k\}$ is a realization of independent Gaussian noise with variance 0.01. Our model for state transitions (3) uses $Q_k = 0.01\ I$ for $k > 1$, and so is identical to the model used to simulate the ground truth $\{x_k\}$. Thus, we have precise knowledge of the process that generated the ground truth $\{x_k\}$. The initial state x_0 is imprecisely specified by setting $g_1(x_0) = (0.1, -0.4)^T \neq x_0$ with corresponding variance $Q_1 = 0.1\ I$.

For $k = 1, \ldots, N$ the measurements z_k were simulated by $z_k = x_{1,k} + v_k$. The measurement noise v_k was generated as follows:

$$v_k \sim (1 - p)\mathbf{N}(0, 1.0) + p\mathbf{N}(0, \phi) . \tag{27}$$

This was done for $p \in \{0, 0.1, 0.2, 0.3\}$ and $\phi \in \{10, 100, 1000\}$. The model for the mean of z_k given x_k is $h_k(x_k) = (1, 0)x_k = x_{1,k}$. As in the previous simulation, we simulated a lack of knowledge of the distribution for the outliers; i.e., $p\mathbf{N}(0, \phi)$. In (3), the model for the variance of z_k given x_k is $R_k = 1.0$.

We simulated 1000 realizations of the ground truth state sequence $\{x_k\}$ and the corresponding measurement sequence $\{z_k\}$. For each realization, we computed the corresponding state sequence estimate $\{\hat{x}_k\}$ using both the IGS and IKS procedures. The MSE corresponding to such an estimate is defined by (25), where x_k is given by (26). For this problem, the model functions $\{g_k(x_{k-1})\}$ are nonlinear so the GKF and RKF were not included. The results of the simulation appear in Table II. As the proportion and variance of the outliers increase, the Gaussian smoother degrades, but the ℓ_1-Laplace smoother is not affected.

Fig. 3 provides a visual illustration of one realization $\{x_k\}$ and its corresponding estimates $\{\hat{x}_k\}$. The left two panels

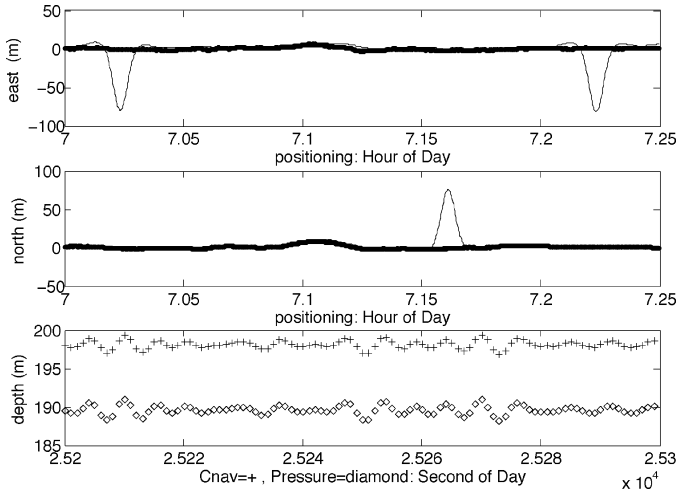

Fig. 4. Track: independent GPS verification (thick line and +), iterated Gaussian smoother estimate (thin line).

demonstrate that, when no outliers are present, both the IGS and ILS generate accurate estimates. Note that we only observe the first component of the state and that the variance of the observation is relatively large (see top two panels). The right two panels show what can go wrong when outliers are present. The Van der Pol oscillator can have sharp peaks as a result of the nonlinearity in its process model, and outliers in the measurements can "trick" the IGS into these modes when they are not really present. In contrast, the Iterated ℓ_1-Laplace Smoother avoids this problem.

VII. Underwater Tracking Application

As a test of the ℓ_1-Laplace smoother, we compared it to an outlier removal method that was used for an underwater tracking experiment. In this experiment, an object was tracked using ocean floor transponders. The object was hung on a steel cable approximately 200 meters below a ship. The ship was pitching and rolling on the surface of the ocean and the pilot of the ship was attempting to "hold station"; i.e., stay at a specific latitude and longitude. A pressure sensor was mounted on the object and it recorded pressure measurements at an approximate rate of once per second. Four acoustic transponders were mounted on the bottom of the ocean and their locations were determined to sub-meter accuracy prior to this experiment. The acoustic travel time between these bottom mounted transponders and the object at the end of the cable was measured at approximately 16-s intervals. The acoustic travel times, and the pressure measurements, were used to estimate the location of the object at the end of the cable.

In addition to the pressure and transponder data, a GPS antenna was located near the support point for the cable and the corresponding latitude, longitude, and height were recorded once per second. The GPS used real time differential corrections to obtain sub-meter accuracy in its positioning. Therefore, the GPS data can be used as an independent verification of the tracking results. In Figs. 4 and 5, the GPS data is denoted using the acronym *Cnav* which is the company that makes the GPS device.

East, north, and depth coordinates are in meters and relative to an origin located at 21.750838 degrees of latitude, 126.148058 degrees of longitude, and mean sea level. We use b_j to denote

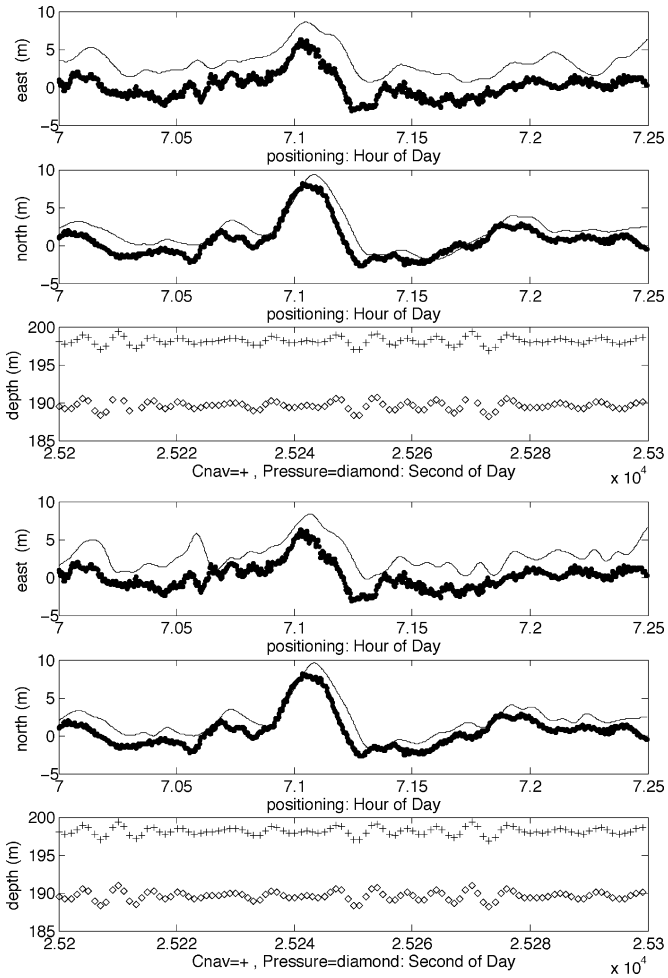

Fig. 5. Track: independent GPS verification (thick line and +), top: Gaussian smoother with outlier removal (thin line), bottom: ℓ_1-Laplace smoother (thin line).

the location of the bottom mounted transponders in (east, north, depth) which were given by

$$\begin{aligned} b_1 &= (128.2, -4655.4,\ 5469.9) \\ b_2 &= (-3293.0,\ 135.8,\ 5472.2) \\ b_3 &= (209.1,\ 4848.4,\ 5424.4) \\ b_4 &= (5336.2,\ 43.5,\ 5318.7). \end{aligned}$$

The speed of sound in the ocean was modeled as constant in the horizontal and varying with depth. The speed as a function of depth $s(u)$ was measured using a CTD which measures water characteristics such as salinity, temperature, pressure, and density. The average sound speed used for the jth bottom mounted transponder was $(b_{3,j} - 200) \Big/ \int_{200}^{b(3,j)} s(u)^{-1}\mathbf{d}u$, where $b_{3,j}$ is the third component of b_j. The resulting sound speed approximations were 1509.2, 1509.2, 1508.9, and 1508.1 meters per second. A delay of .003 seconds was subtracted from the round trip travel time measurements, then they were divided by two and multiplied by the sound speed for the corresponding transponder to get ranges. A range correction was added to this straight line approximation so that the travel time matched acoustic ray tracing at a nominal position. These range corrections were approximately $\Delta r_1 = 0.4$, $\Delta r_2 = 0.1$, $\Delta r_3 = 0.4$, and $\Delta r_4 = 0.6$, meters.

Pressure measurements in absolute bars were converted to depth in meters (below the surface of the ocean) by the formula depth $= 9.9184(\text{pressure} - 1)$.

We use N to denote the total number of time points at which we have tracking data. For $k = 1, \ldots, N$, the state vector at time t_k is defined by $x_k = (e_k, n_k, d_k, \dot{e}_k, \dot{n}_k, \dot{d}_k)^T$ where (e_k, n_k, d_k) is the (east, north, depth) location of the object (in meters from the origin), and $(\dot{e}_k, \dot{n}_k, \dot{d}_k)$ is the time derivative of this location.

The measurement vector at time t_k is denoted by z_k. The first four components of z_k are the range measurements to the corresponding bottom mounted transponders and the last component is the depth corresponding to the pressure measurement. For $j = 1, \ldots, 4$, the model for the mean of the corresponding range measurements was $h_{j,k}(x_k) = ||(e_k, n_k, d_k) - b_j||_2 - \Delta r_j$. Note that the functions $h_{j,k}$ are nonlinear due to the presence of the norm. These measurements were modeled as independent and having a standard deviation of 3 meters. The model for the mean of the pressure measurement was $h_{5,k}(x_k) = d_k$. These measurements were modeled as having a standard deviation of 0.05 meters (the pressure sensor was much more accurate than the range measurements).

We use Δt_k to denote $t_{k+1} - t_k$. The model for the mean of x_{k+1} given x_k; i.e., $g_{k+1}(x_k)$ is equal to

$$(e_k + \dot{e}_k \Delta t_k,\ n_k + \dot{n}_k \Delta t_k,\ d_k + \dot{d}_k \Delta t_k,\ \dot{e}_k,\ \dot{n}_k,\ \dot{d}_k)^T .$$

The east, north, and depth components of x_{k+1} given x_k are modeled as normally distributed with mean zero and standard deviation $.01\Delta t_k$. The derivative of east, north, and depth components of x_{k+1} given x_k are modeled as normally distributed with mean zero and standard deviation $.2\Delta t_k$.

The iterated Kalman smoother implemented in the ckbs-07–11-07 package uses a Gaussian model for the measurement errors. This Gaussian smoother allows for removing measurements by setting the corresponding inverse variance to zero. It was used to estimate the state sequence $\{x_k\}$, and the corresponding residuals for each component of $z_k - h_k(x_k)$ were computed. These residuals were used to compute a sample standard deviation for each of the measurement components. Measurement components that corresponded to more than three times the sample standard deviation were removed and the estimate for the state sequence was recalculated with the subsampled data set.

To illustrate the need for outlier removal, the Gaussian smoother results without outlier removal are shown in Fig. 4. There are three large peaks (two in the east component and one in the north component of the state) that appear as a result of large measurement noise.

The two fits are shown side by side in Fig. 5. The darker curves appearing below the track are independent verifications using the GPS tracking near the top of the cable. The pressure sensor was approximately 198 meters below the surface and the GPS antenna was approximately 8 meters above the surface during this comparison. A depth of 198 meters was added to the depth location of the GPS antenna so that the depth comparison can use the same axis for both the GPS data and the tracking results (and the difference should be about 8 meters). The time

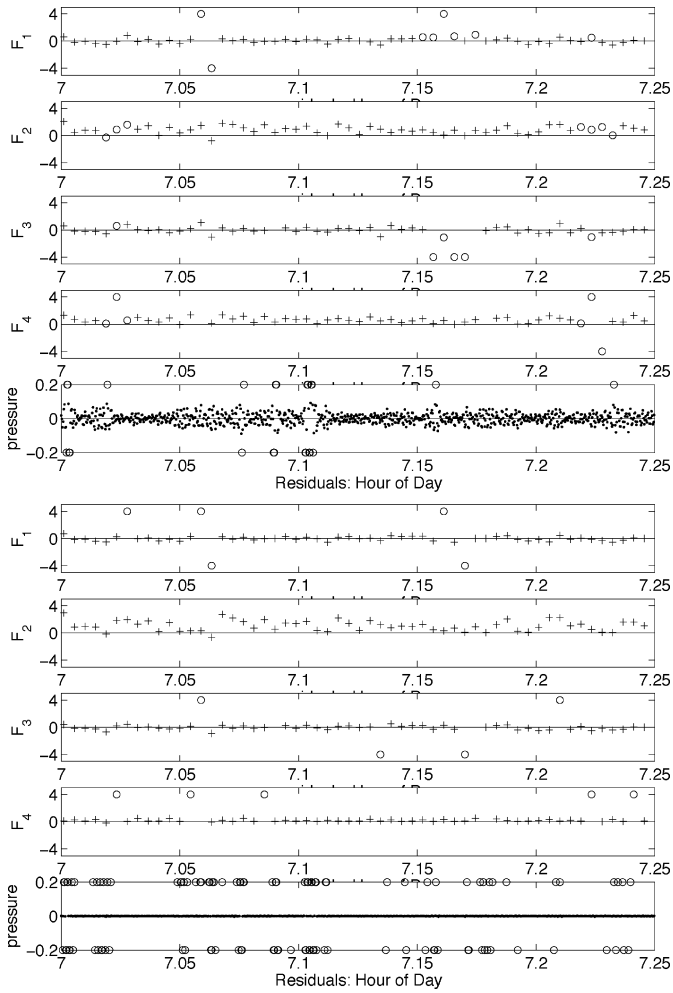

Fig. 6. Residuals top: Gaussian smoother with outlier removal, bottom: ℓ_1-Laplace smoother. In these plots, transponders are labeled by their frequencies in kHz; $F_1 = 11.25$, $F_2 = 11.75$, $F_3 = 12.25$, and $F_4 = 12.75$. All residuals are in meters.

scale for the depth plots is much finer (a total of a hundred seconds instead of one quarter hour) and demonstrates the accuracy of the GPS tracking as validated by the pressure sensor. The ℓ_1-Laplace smoother was able to use the whole data sequence, despite large outliers in the data. The fits look very similar, and it is clear that the ℓ_1-Laplace smoother is not affected by large outliers in the data.

It is interesting to compare the residuals. These are presented side by side in Fig. 6. Outliers are shown as "o" characters in this figure and outliers with absolute value larger than three standard deviations are plotted at the vertical axis limits. It is important to note that the Gaussian smoother with outlier removal detects outliers after the first fit that are not outliers after the second fit. This results in "over-smoothing" of the outlier removal track and more detail in the ℓ_1-Laplace track in Fig. 5.

The ℓ_1-Laplace smoother pushes more of the residuals to zero, which we expect. In particular, the residuals corresponding to the depth measurements (the most frequent and accurate measure) are often zero for the ℓ_1-Laplace smoother. It is as if the ℓ_1-Laplace smoother detects which pressure measurements are most accurate and treats those measurements as constraints during the fitting.

VIII. Conclusion

This paper presents an efficient method for solving the MAP estimation problem for the Kalman smoother with ℓ_1-Laplace measurement noise. The ℓ_1-Laplace smoother was demonstrated to have lower mean squared error than the iterated Kalman smoother across several simulated outlier schemes for both linear and nonlinear process models. In addition, the ℓ_1-Laplace smoother did not have the "over-smoothing" problem that occurred when a common outlier removal scheme was applied to real-world underwater tracking data. The techniques presented here can be extended to other robust measurement noise distributions and to robustness with respect to the noise in the dynamics. Conceptually, a solution to the problem treated in this paper that takes into account the shape of the posterior of the states could be obtained using particle filters and Monte Carlo smoothing approaches, see, e.g., [2], [12], [14], and [17]. These techniques are also able to return confidence intervals around the estimates. However, already in the Gaussian case they require a delicate tuning of the proposal densities necessary to obtain the posterior in sampled form, see, e.g., [22], a problem which can be further complicated by the presence of Laplace priors (see the discussion in [43, subsec. 8.3]). It remains an open question for future research as to whether the methods proposed in this paper can also be used to provide efficient proposal densities for robust Markov chain Monte Carlo smoothing.

Acknowledgment

The authors would like to thank the North Pacific Acoustic Laboratory (NPAL) investigators of the Applied Physics Laboratory, University of Washington, for the underwater tracking data used in this paper (NPAL is sponsored by the Office of Naval Research code 321OA).

References

[1] A. Aravkin, "Robust methods for Kalman filtering/smoothing and bundle adjustment," Ph.D. dissertation, Univ. of Washington, Seattle, 2010.

[2] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter*. Boston, MA: Artech House, 2004.

[3] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: Wiley, 2001.

[4] B. Bell, "The iterated Kalman smoother as a Gauss–Newton method," *SIAM J. Optim.*, vol. 4, no. 3, pp. 626–636, Aug. 1994.

[5] B. M. Bell, "The marginal likelihood for parameters in a discrete Gauss-Markov process," *IEEE Trans. Signal Process.*, vol. 48, no. 3, pp. 626–636, Mar. 2000.

[6] B. M. Bell, J. V. Burke, and G. Pillonetto, "An inequality constrained nonlinear Kalman-Bucy smoother by interior point likelihood maximization," *Automatica*, pp. 25–33, 2008.

[7] B. M. Bell and G. Pillonetto, Matlab®/Octave Package, 2007 [Online]. Available: http://www.seanet.com/~bradbell/ckbs/ckbs.xml

[8] G. E. P. Box, R. L. Launer and G. N. Wilkinson, Eds., "Robustness in the strategy of scientific model building," in *Robustness Statist.: Workshop Proc.*, 1979.

[9] J. V. Burke, "Descent methods for composite nondifferentiable optimization problems," *Math. Program.*, vol. 33, pp. 260–279, 1985.

[10] J. V. Burke, "Second order necessary and sufficient conditions for convex composite ndo," *Math. Program.*, vol. 38, pp. 287–302, 1987.

[11] A. E. Cetin and A. M. Tekalp, "Robust reduced update Kalman filtering," *IEEE Trans. Circuits Syst.*, vol. 37, no. 1, pp. 155–156, Jan. 1990.

[12] R. Chen and J. S. Liu, "Mixture Kalman filters," *J. R. Statist. Soc. Ser. B*, vol. 62, pp. 493–508, 2000.

[13] T. Cipra and R. Romera, "Kalman filter with outliers and missing observations," *Sociedad de Estadistica e Invastigacion Operativa*, vol. 6, no. 2, pp. 379–395, 1997.

[14] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer, 2001.

[15] J. Durbin and S. J. Koopman, "Monte Carlo maximum likelihood estimation for non-Gaussian state space models," *Biometrika*, vol. 84, pp. 669–684, 1997.

[16] Z. M. Durovic and B. D. Kovachevic, "Robust estimation with unknown noise statistics," *IEEE Trans. Autom. Control*, vol. 44, no. 6, pp. 1292–1296, Jun. 1999.

[17] P. Fearnhead and P. Clifford, "On-line inference for hidden markov models via particle filters," *J. R. Statist. Soc. Ser. B*, vol. 65, pp. 887–899, 2003.

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

[19] V. Gebski and D. McNeil, "A refined method of robust smoothing," *J. Amer. Statist. Assoc.*, vol. 79, no. 387, pp. 616–623, Sep. 1984.

[20] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov Chain Monte Carlo in Practice*. London, U.K.: Chapman & Hall, 1996.

[21] S. Gillijns, O. B. Mendoza, J. Chandrasekar, B. L. R. De Moor, D. S. Bernstein, and A. Ridley, "What is the ensemble Kalman filter and how well does it work?," in *Proc. Amer. Control Conf.*, 2006, pp. 4448–4453.

[22] S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing for nonlinear time series," *J. Amer. Statist. Assoc.*, vol. 99, pp. 156–168, 2004.

[23] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics*. : Wiley, 1986.

[24] W. K. Hastings, "Monte Carlo sampling methods using Markov chain and their applications," *Biometrika*, vol. 57, pp. 97–109, 1970.

[25] G. A. Hewer, R. D. Martin, and J. Zeh, "Robust preprocessing for Kalman filtering of glint noise," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-23, no. 1, pp. 120–128, Jan. 1987.

[26] P. J. Huber, *Robust Statistics*. New York: Wiley, 2004.

[27] B. Jungbacker and S. J. Koopman, "Monte Carlo estimation for nonlinear non-Gaussian state space models," *Biometrika*, vol. 94, pp. 827–839, 2007.

[28] R. Kandepu, B. Foss, and L. Imsland, "Applying the unscented Kalman filter for nonlinear state estimation," *J. Process Control*, vol. 18, pp. 753–768, 2008.

[29] S. A. Kassam and H. V. Poor, "Robust techniques for signal processing: A survey," *Proc. IEEE*, vol. 73, no. 3, pp. 433–481, Mar. 1985.

[30] R. L. Kirlin and A. Moghaddamjoo, "Robust adaptive Kalman filtering for systems with unknown step inputs and non-Gaussian measurement errors," *IEEE Trans. Acoust. , Speech, Signal Process.*, vol. ASSP-34, no. 2, pp. 252–263, Mar. 1986.

[31] S. Kotz, T. J. Kozubowski, and K. Podgorski, *The Laplace Distribution and Generalizations*. Cambridge, MA: Birkhauser, 2001.

[32] J. S. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems," *J. Amer. Statist. Assoc.*, vol. 93, pp. 1032–1044, 1998.

[33] R. A. Maronna, D. Martin, and Yohai, *Robust Statistics*. New York: Wiley, 2006, Wiley Series in Probability and Statistics.

[34] C. J. Masreliez and R. D. Martin, "Robust Bayesian estimation for the linear model and robustifying the Kalman filter," *IEEE Trans. Autom. Control*, vol. AC-22, no. 3, pp. 361–371, Jun. 1977.

[35] R. J. Meinhold and N. D. Singpurwalla, "Robustification of Kalman filter models," *J. Amer. Statist. Assoc.*, vol. 84, no. 406, pp. 479–486, Jun. 1989.

[36] K. Murphy, Matlab® Toolbox for Kalman Filtering, 1998, [Online]. Available: http://www.cs.ubc.ca/~murphyk/Software/Kalman/kalman.html

[37] I. R. Petersen and A. V. Savkin, *Robust Kalman Filtering for Signals and Systems with Large Uncertainties*. Cambridge, MA: Birkhauser, 1999.

[38] G. Pillonetto and B. M. Bell, "Optimal smoothing of non-linear dynamic systems via Monte Carlo Markov chains," *Automatica*, vol. 44, pp. 1676–1685, 2008.

[39] R. T. Rochafellar, *Convex Analysis*. Princeton, NJ: Princeton Univ. Press, 1970.

[40] R. T. Rochafellar and R. J. B. Wets, *Variational Analysis*. New York: Springer, 1998.

[41] P. Ruckdechel and B. Spangl, An R-Package for Robust Kalman Filtering, 2008, [Online]. Available: http://robkalman.r-forge.r-project.org/

[42] I. C. Schick and S. K. Mitter, "Robust recursive estimation in the presence of heavy-tailed observation noise," *Ann. Statist.*, vol. 22, no. 2, pp. 1045–1080, Jun. 1994.

[43] M. W. Seeger, "Bayesian inference and optimal design for the sparse linear model," *J. Mach. Learn. Res.*, vol. 9, pp. 759–813, 2008.

[44] J. C. Spall, "Estimation via Markov chain Monte Carlo," *IEEE Control Syst. Mag.*, vol. 23, no. 2, pp. 34–45, Apr. 2003.

Aleksandr Y. Aravkin was born in St. Petersburg, Russia. He received the B.S. degree in mathematics and computer science, the M.S. degree in statistics, and the Ph.D. degree in mathematics from the University of Washington, Seattle.

He currently holds a postdoctoral appointment in the Department of Earth and Ocean Sciences, University of British Columbia, Vancouver, BC, Canada. His interests include robust statistical modeling, numerical optimization, and algorithm design, with applications to Kalman filtering/smoothing, computer vision, and large-scale inverse problems such as PDE constrained optimization.

Bradley M. Bell was born in New York in 1952. He received the B.A. degree in math and physics from Saint Lawrence University, Canton, NY, in 1973 and the M.A. and Ph.D. degrees in mathematics from the University of Washington, Seattle, in 1976 and 1984, respectively.

Currently, he is with the Applied Physics Laboratory, University of Washington. Much of his current work is in collaboration with the UW School of Public Health as a member of The Institute for Health Metrics and Evaluation. He is also the project manager for the Coin-OR Algorithmic Differentiation project CppAD. His current research is focused on statistically motivated numerical methods for scientific modeling and data analysis. This includes optimization, Monte-Carlo simulation, function estimation (Tikhonov regularization), nonlinear mixed effects modeling, Kalman filtering and smoothing, and algorithmic differentiation.

James V. Burke was born in New Jersey in 1955. He received the B.S. degree from Knox College, Galesburg, IL. in 1977 and the Ph.D. degree in mathematics from the University of Illinois at Urbana-Champaign in 1983.

He was an Assistant Professor of mathematics at the University of Kentucky, Lexington, from 1983 to 1985. He then moved to the University of Washington in 1985 where he is now a Professor of mathematics holding adjunct positions in applied mathematics, statistics, and bioengineering. He is past Director of the Applied and Computational Mathematical Sciences Program and past Assistant Director at the Resource Facility for Population Kinetics at the University of Washington. Research interests include numerical, theoretical, and applied optimization and variational analysis with a current focus on applications in statistics, new approaches to Kalman smoothing, eigenvalue optimization, and stability theory.

Gianluigi Pillonetto was born in Montebelluna (TV), Italy, on January 21, 1975. He received the Ph.D. degree (*cum laude*) in computer science engineering from the University of Padua, Padua, Italy, in 1998 and the Ph.D. degree in bioengineering from the Polytechnic of Milan, Milan, Italy, in 2002.

In 2000 and 2002, he was visiting scholar and visiting scientist, respectively, at the Applied Physics Laboratory, University of Washington, Seattle. From 2002 to 2005, he was a Research Associate in the Department of Information Engineering, University of Padua. Since 2005, he has been an Assistant Professor of control and dynamic systems in the Department of Information Engineering, University of Padua. His research interests are in the fields of system identification, stochastic systems, deconvolution problems, nonparametric regularization techniques, learning theory, and randomized algorithms.