Robust Singular Smoothers For Tracking Using Low-Fidelity Data

Jonathan Jonker*, Aleksandr Aravkin†, James V. Burke*, Gianluigi Pillonetto‡ and Sarah Webster§

*Department of Mathematics, University of Washington, Seattle, WA USA, *jonkerjo@uw.edu, jvburke@uw.edu*
†Department of Applied Mathematics, University of Washington, Seattle, WA USA, *saravkin@uw.edu*
‡Department of Information Engineering, University of Padova, Italy, *giapi@dei.unipd.it*
§Applied Physics Laboratory, University of Washington, Seattle, WA USA, *swebster@apl.washington.edu*

Abstract—**Tracking underwater autonomous platforms is often difficult because of noisy, biased, and discretized input data. Classic filters and smoothers based on standard assumptions of Gaussian white noise break down when presented with any of these challenges. Robust models (such as the Huber loss) and constraints (e.g. maximum velocity) are used to attenuate these issues. Here, we consider robust smoothing with singular covariance, which covers bias and correlated noise, as well as many specific model types, such as those used in navigation. In particular, we show how to combine singular covariance models with robust losses and state-space constraints in a unified framework that can handle very low-fidelity data. A noisy, biased, and discretized naviga- tion dataset from a submerged, low-cost inertial measurement unit (IMU) package, with ultra short baseline (USBL) data for ground truth, provides an opportunity to stress-test the proposed framework with promising results. We show how robust modeling elements improve our ability to analyze the data, and present batch processing results for 10 minutes of data with three different frequencies of available USBL position fixes (gaps of 30 seconds, 1 minute, and 2 minutes). The results suggest that the framework can be extended to real-time tracking using robust windowed estimation.**

I. INTRODUCTION

State-space models are ubiquitous in signal processing, and allow integration of disparate measurements to inform esti- mation, decisions, and control. Classic filtering [1, 2] and smoothing [3, 4] are core tools used to estimate these models. Their dependence on high-fidelity data, driven by Gaussian assumptions on errors and innovations, has demanded unequiv- ocal attention from researchers and practitioners, and inspired robust dynamic inference methods.

While early robust approaches [5, 6] sought to modify itera- tions of the Kalman filter (KF) and Rauch-Tung-Striebel (RTS) smoother, over the last 25 years researchers have used *robust formulations* to weave assumptions on errors and innovations directly into the estimation problems themselves [7, 8, 9, 10, 11, 12, 13]. Constraints, when available, are also readily incorporated into the problem formulation [14]. Specifying the formulation leaves one free to choose from a range of optimization algorithms; the survey [15] describes a general class of models as well as first- and second-order methods to solve them.

Our focus is on models with singular variances for process and measurement residuals. These models are excluded by

the assumptions of generalized smoothing [15] and all of the various special cases cited in the survey. In this paper, we build on the recently proposed framework of [16] for singular models, and systematically develop complementary modeling elements: **robust penalties, informative constraints, and singular models**. The resulting approach exploits the structure of singular covariances head on rather than using workarounds such as pseudo-inverses or variance boosting that either do not work in the general setting or introduce unnecessary changes to the fundamental model (see discussion in [16]). A simple
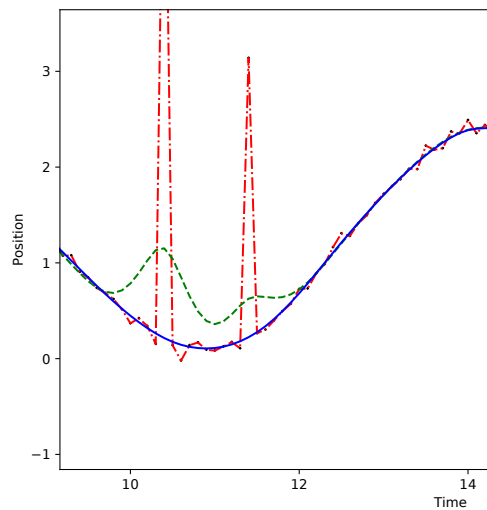

Fig. 1: We track a simple trajectory in the presence of outliers. The red dash-dot shows a 'robust' Huberized approach implemented using a pseudo-inverse; green dash shows the proposed singular ℓ^2 estimate; blue solid shows the proposed singular Huber estimate, which clearly tracks the true state.

synthetic tracking example with a singular process model shows how the common tack of replacing the inverse by the pseudo-inverse fails dramatically in the presence of outliers (Figure 1). Any robust approach requires control of the null spaces associated to process and observations. We develop a direct practical formulation and method, and test it on a batch smoothing analysis of real field data, with a view towards real- time implementation in future work.

Background. Our main goal is to infer an unobserved state sequence $x_1, \ldots, x_N$ from noisy observations $y_1, \ldots, y_N$ using the model:

$$
\begin{aligned}
x_1 &= x_0 + w_1 \\
x_k &= G_k x_{k-1} + w_k, \quad k = 2, \ldots, N \\
y_k &= H_k x_k + v_k, \quad k = 1, \ldots, N, \\
x_k &\in X_k, \quad \text{each } X_k \text{ polyhedral},
\end{aligned}
\tag{1}
$$

where x_0 is a given initial state estimate, G_k and H_k are linear process and measurement models, $y_1, \ldots, y_N$ are observations, and X_k specify additional information through constraints. The framework of [15] assumes that w_k and v_k are mutually independent random variables with known *nonsingular* covariances Q_k and R_k, and that they follow from log-concave distributions; in particular they may be non-Gaussian.

Synthesizing all of this information gives the problem

$$
\begin{aligned}
\min_{x_1 \in X_1, \ldots, x_n \in X_N} \sum_k & \rho_p(Q_k^{-1/2}(G_k x_k - x_{k-1})) \\
& + \rho_m(R_k^{-1/2}(H_k x_k - y_k)),
\end{aligned}
\tag{2}
$$

with ρ_p and ρ_m convex penalties. Using (2) provides estimates that are robust to outliers and can follow sudden changes in the state. Most of the inference- or optimization-based work in the convex dynamic setting is a special case of (2). Many examples, including robust penalties and constraints, are collected in [15].

We now extend to *singular covariances* R_k (for errors v_k) and Q_k (for innovations w_k). These models specify key use cases, particularly for innovations (process) modeling, briefly summaried below (see [16] for a more detailed discussion).
Deterministic integrals. Most models in robotics, particularly in navigation, use integration to model process relationships between state variables (e.g. when position, velocity, and acceleration are part of the state). Any deterministic integral yields a singular process model. The simplest example (with position a direct integral of stochastic velocity) is used to create Figure 1.
Nuisance parameters. Unknown constants that need calibration (such as fixed instrument biases) require special modeling in the nonsingular paradigm (2). With singular models, we can augment the state in order to infer these parameters.
Auto-regressive models and correlated errors. State-space models are broadly used in auto-regressive, moving average, and time series models [17]. These elements also appear in general smoothing models, particularly to deal with correlated measurement errors [18].

All three examples are accessible in the classic linear Gaussian setting. The KF need not invert Q or R, and provides the minimum variance estimate for both singular and nonsingular models [19]. Some algorithms rely on precise knowledge of the error structure or explicit equality constraints [20, 21, 22]. Correlated errors are dealt with by augmenting the state and using a singular model [18]. **None of these techniques generalize to singular models in the setting of** (2), and some naive generalizations fail dramatically (Figure 1).

This paper builds on the reformulation of [16] for singular models. We develop a systematic approach and test it using a navigation model with a real-world mooring dataset. We show how robust statistics, singular models, and constraints can be systematically used to overcome a range of challenges simultaneously present in the dataset: (1) outliers, (2) deterministic relationships between states, (3) measurement biases, and (4) coarsely discretized observations.

The paper proceeds as follows. Section II summarizes the singular formulation of [16] and relevant optimization algorithms. Section III develops the key modeling elements to address common data challenges. Section IV presents the navigation models. Section V shows how the model elements come together to analyze the target mooring dataset, and obtain a high fidelity track from low-fidelity observations. Section VI concludes with discussion and future work.

II. ROBUST SINGULAR FORMULATION AND ALGORITHM

We reformulate the robust smoothing problem to seamlessly allow both singular and nonsingular covariance models for errors and innovations. The resulting problem can be solved with any primal-dual algorithm. We show how to apply the classic Douglas-Rachford splitting (DRS) algorithm (see e.g. [23, 24]) to the reformulated problem.

Problem (2) can be reformulated by introducing auxiliary variables u_k and t_k to represent pre-whitened innovations and measurement residuals:

$$
\begin{aligned}
\min_{x,u,t} \quad & \sum_k \rho_p(u_k) + \rho_m(t_k) + \rho_s(x_k) \\
\text{s.t.} \quad & Q_k^{1/2} u_k = G_k x_k - x_{k-1} \\
& R_k^{1/2} t_k = H_k x_k - y_k
\end{aligned}
\tag{3}
$$

where $\rho_s(x_k)$ may be taken as the convex indicator function to recover the constraints in (2):

$$
\rho_s(x_k) = \begin{cases} 0 & x_k \in X_k \\ \infty & x_k \notin X_k. \end{cases}
$$

When Q_k and R_k are invertible, we can solve for u_k, t_k and recover (2). Otherwise, problem (3) is well-posed while (2) is not. We can write (3) in compact form

$$
\min_z \quad \rho(z) \quad \text{s.t. } Az = \hat{w},
$$

$$
\rho(z) = \sum_{k=1}^N \rho_p(u_k) + \rho_m(t_k) + \rho_s(x_k).
\tag{4}
$$

where

$$
\begin{aligned}
z^T &= \begin{pmatrix} u_1^T & t_1^T & x_1^T & \ldots u_N^T & t_N^T & x_N^T \end{pmatrix} \\
\hat{w}^T &= \begin{pmatrix} x_0^T & y_1^T & 0 & y_2^T & \ldots & 0 & y_N^T \end{pmatrix},
\end{aligned}
\tag{5}
$$

$$
A = \begin{pmatrix} D_1 & 0 & \ldots & & 0 \\ B_1 & D_2 & 0 & & \vdots \\ 0 & \ddots & \ddots & & 0 \\ 0 & 0 & & B_{N-1} & D_N \end{pmatrix},
\tag{6}
$$

and

$$D_i = \begin{pmatrix} Q_i^{1/2} & 0 & I \\ 0 & R_i^{1/2} & H_i \end{pmatrix}, B_j = \begin{pmatrix} 0 & 0 & -G_{j+1} \\ 0 & 0 & 0 \end{pmatrix}.$$

The variables are ordered in such a way that A is block bi-diagonal. If all observations z_i lie in the range of H_i, the constraint $Az = \hat{w}$ will be feasible [16].

The problem (4) is a convex optimization problem and can be solved using a variety of techniques. We show that the DRS algorithm is straightforward to implement, and preserves the computational complexity of the classic KF/RTS algorithms because of the way A is structured in (6).

Given a convex function f, its convex conjugate f^* is given by

$$f^*(y) = \sup_x \langle x, y \rangle - f(x),$$

and its proximal operator with step α, denoted by $\mathrm{prox}_{\alpha f}$ (see e.g. [25]) is given by:

$$\mathrm{prox}_{\alpha f}(\zeta) = \arg\min_x \frac{1}{2\alpha} \|\zeta - x\|^2 + f(x). \tag{7}$$

For a long list of objectives, prox operators are available in closed form or are efficiently computable. In particular this is the case when ρ_p, ρ_m, ρ_s form any subset of the numerous elements briefly surveyed in Section III. It is actually the prox of ρ^* that appears in the DRS iteration (Algorithm 1) rather than the prox of ρ, but these are linked by the simple formula

$$\mathrm{prox}_\rho(z) + \mathrm{prox}_{\rho^*}(z) = z.$$

To specify the algorithm, we let $g(z)$ be the indicator of the affine feasible region $Az = \hat{w}$:

$$g(z) = \begin{cases} 0 & Az = \hat{w} \\ \infty & Az \neq \hat{w} \end{cases}$$

Problem (4) can now be written simply as

$$\min_z \rho(z) + g(z)$$

which is a natural template for DRS, detailed in Algorithm 1.

Algorithm 1 Douglas-Rachford Splitting (DRS)

Input: Initialize at any z^0, ζ^0.
1: **loop**
2: $\quad z^k = \mathrm{prox}_{\tau g}(z^{k-1} - \tau \zeta^{k-1})$
3: $\quad \zeta^k = \mathrm{prox}_{\sigma \rho^*}(\zeta^{k-1} + \sigma(2z^k - z^{k-1}))$
$\quad$ **return** z^k

To implement Algorithm 1 we need proximal operators of $\rho_p, \rho_m,$ and ρ_s. Eight common piecewise linear-quadratic (PLQ) penalties are shown in Figure 2, and their proximal operators are summarized in Table I.

The proximal operator for g is given by

$$\mathrm{prox}_g(\eta) = \underset{Az=\hat{w}}{\mathrm{argmin}} \frac{1}{2} \|\eta - z\|^2$$

(a) quadratic $\qquad$ (b) 1-norm

(c) quantile, $\tau = 0.3$ $\qquad$ (d) huber, $\kappa = 1$

(e) quantile huber $\qquad$ (f) vapnik, $\epsilon = 0.5$
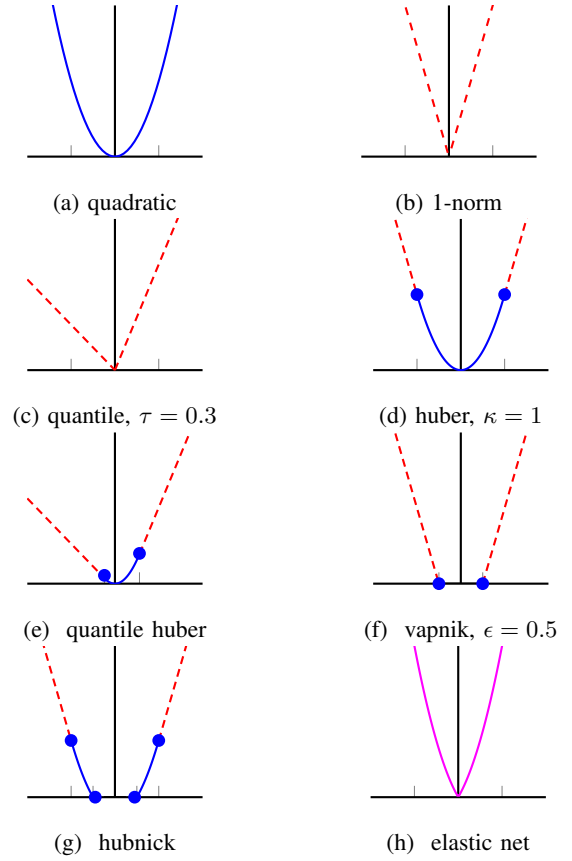
(g) hubnick $\qquad$ (h) elastic net

Fig. 2: Common piecewise linear-quadratic (PLQ) losses.

which is a least squares problem with affine constraints. Solving it efficiently leverages the structure of (6). In particular we need to solve a single structured linear system

$$\begin{bmatrix} I & A^T \\ 0 & AA^T \end{bmatrix} \begin{bmatrix} z \\ \nu \end{bmatrix} = \begin{bmatrix} \eta \\ A\eta - \hat{w} \end{bmatrix} \tag{8}$$

where AA^T is block tridiagonal and does not change between iterations. In our implementation, we need only compute a single block bidiagonal factorization once, which can then be used to solve (8) in $O(n^2 N)$ operations in each iteration, no more expensive than a single matrix-vector multiply.

For piecewise-linear quadratic ρ [26, 15], DRS converges to an optimal solution at a local linear rate [16], which does not depend on the condition number of A. A good initialization makes DRS competitive with the fastest available solvers, even second order methods with quadratic local rates [15].

III. MODELING ELEMENTS

The proposed framework has three complementary modeling elements: singular covariance matrices Q and R; process/measurement penalties ρ_p, ρ_m; and constraints ρ_s on the state. In this section, we show a range of choices for each element, and compute the operators required for Algorithm 1. **Singular covariances** can be used to capture affine constraints, auto-regressive structure, integrated errors, and bias.

- *Affine constraints using singular R.* the ith element of the state at time k is known exactly, add row

$$\begin{bmatrix} 0 & \dots & 0 & \underbrace{1}_{i} & 0 & \dots & 0 \end{bmatrix}$$

to the measurement model H_k, a row and column of zeros to R_k, and the known value as the last element of z_k.
- *Bias with singular Q.* A common model for bias is to include it as a non-varying component across the state:

$$\widetilde{x}_k = \begin{bmatrix} x_k \\ b \end{bmatrix}, \quad \widetilde{Q}_k = \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix}.$$

- *Correlated noise using singular Q.* Correlated noise w_k is typically modeled by [27]

$$w_k = M w_{k-1} + \beta_k, \quad \beta_k \sim N(0, Q).$$

Here too we can augment the state and use a singular process variance:

$$\widetilde{x}_k = \begin{bmatrix} x_k \\ w_k \end{bmatrix}, \widetilde{G}_k = \begin{bmatrix} G_k & I \\ 0 & M \end{bmatrix}, \quad \widetilde{Q}_k = \begin{bmatrix} 0 & 0 \\ 0 & Q_k \end{bmatrix}.$$

Piecewise linear-quadratic (PLQ) Penalties. The proposed framework allows process innovations, measurement residuals, and state regularization to come from any convex prox-friendly penalty. To keep the exposition simple, we collect eight commonly used convex piecewise linear-quadratic penalties in Figure 2, and compute their prox operators in Table I. The penalties can be thought of in terms of three features:

- Behavior at origin: nonsmooth features encourage exact fitting of the quantity being measured, while deadzones are appropriate for discretized observations.
- Tail growth: asymptotically linear penalties are more tolerant of large inputs. Applied to measurements, this gives robustness to outliers; applied to innovations, it gives an ability to quickly track evolving trends.
- Asymmetry: allows handling of special cases where underestimating is qualitatively different from over-estimating.

Constraints. It is very convenient to enforce simple constraints on the state estimates x_k. If we take $\rho_s(x) = \delta_X(x)$ then the prox operator prox_{ρ_s} is simply the projection onto the set X. Box constraints are a very common type of constraints that enforce known bounds on the state, and have a trivial projection. The proposed framework allows us to use any convex region that has a computationally efficient projection.

IV. NAVIGATION MODEL

We use a constant-velocity kinematic model that is appropriate for many underwater vehicle applications, where accelerations are heavily damped and trajectories are often long straight lines (e.g. for transit or survey work). When the attitude is known or changing slowly, the model can be linearized effectively. For a vehicle that is well-instrumented in attitude, the uncertainty in position (and the x-y states in particular) is typically orders of magnitude larger than the uncertainty in attitude. Thus, in practice, we often simplify the full nonlinear

TABLE I: Prox operators of common PLQ penalties.

Penalty f	$\text{prox}_{\alpha f}(z)$	Ref.
$\frac{1}{2}\|x\|^2$, Fig. 2a	$\frac{1}{1+\alpha}z$	[28, 29]
$\|x\|_1$, Fig. 2b	$\text{sign}(z) \odot (\|z\| - \alpha)_+$	[30, 31]
q_τ, Fig. 2c	$\begin{cases} z_i - \alpha(1-\tau) & z_i > \alpha(1-\tau) \\ z_i + \alpha\tau & z_i < -\alpha\tau \\ 0 & \text{else} \end{cases}$	[32, 33]
h_κ, Fig. 2d	$\frac{\alpha}{\alpha+\kappa}z + \frac{\kappa}{\alpha+\kappa}\text{prox}_{(\alpha+\kappa)\|\cdot\|_1}(z)$	[34]
$q_{\tau,\kappa}$, Fig. 2e	$\frac{\alpha}{\alpha+\kappa}z + \frac{\kappa}{\alpha+\kappa}\text{prox}_{(\alpha+\kappa)q_\tau}(z)$	[35]
v_ϵ, Fig. 2f	$\begin{cases} z_i - \alpha & z_i > \epsilon + \alpha \\ \epsilon & \epsilon < z_i \le \alpha + \epsilon \\ z_i & -\epsilon \le z_i \le \epsilon \\ -\epsilon & -\epsilon - \alpha < z_i \le -\epsilon \\ z_i + \alpha\tau & z_i < -\alpha - \epsilon. \end{cases}$	[36]
hubnik-κ, Fig. 2g	$\frac{\alpha}{\alpha+\kappa}z + \frac{\kappa}{\alpha+\kappa}\text{prox}_{(\alpha+\kappa)v_\epsilon}(z)$	[37, 38]
e-net, Fig. 2h	$\text{prox}_{\frac{\alpha}{1+2\alpha}\|\cdot\|_1}\left(\frac{1}{1+2\alpha}z\right)$	[39]

vehicle process model to track only position states (x, y, z), while assuming that the attitude states (r, p, h) are directly available from the most recent sensor measurements. To make the model linear, the position and its derivatives are referenced to the local-level frame.

An effective model must counteract biases, outliers, and data discretization in the IMU data. We develop this model using the elements of the proposed framework.

Process model. To incorporate linear acceleration measurements from an IMU, we track linear velocities and linear acceleration in the state vector:

$$x_s = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z}]^\top. \tag{9}$$

The linear kinematic process model is given by

$$\dot{x}_s = \underbrace{\begin{bmatrix} 0 & I & 0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{bmatrix}}_{F_s} x_s + \underbrace{\begin{bmatrix} 0 \\ I \\ 0 \end{bmatrix}}_{G_s} w_s, \tag{10}$$

where $w_s \sim \mathcal{N}(0, Q_s)$ is zero-mean Gaussian noise. The linear process model (10) is discretized using a Taylor series:

$$x_{s_{k+1}} = F_{s_k} x_{s_k} + w_{s_k} \tag{11}$$

$$F_{s_k} = e^{F_s T} \approx\approx \begin{bmatrix} I & IT & \frac{1}{2}IT^2 \\ 0 & I & IT \\ 0 & 0 & I \end{bmatrix},$$

where I in (11) denotes the 3×3 identity matrix, and the higher order terms are identically zero because of the structure of F_s. We model the process covariance as if the error were the next term in the Taylor series approximation, a technique suggested by [40]. More precisely we set covariance to be the outer product, $\Gamma^T\Gamma$ where

$$\Gamma = \begin{bmatrix} \frac{1}{3!}IT^3 & \frac{1}{2!}IT^2 & IT \end{bmatrix}$$

This leads to a rank 3 covariance for a 9×9 matrix for a model that comprises position, velocity, and acceleration.

Given this covariance structure, the process model will penalize changes in acceleration. As the vehicle travels in a

relatively straight line with small corrections, we expect to see acceleration mostly constant with a few small jumps. We use the ℓ_1 norm for innovations, as it encourages exact fits while simultaneously allowing occasional sudden changes.

Measurement model. The inertial measurement unit (IMU) measures linear and angular accelerations relative to the physical frame of the vehicle on which it is mounted, while the state tracks linear acceleration relative to the navigation frame. We obtain the coordinate transformation between these frames using heading, pitch, and roll of the vehicle:

$$R(\varphi) = R_h^\top R_p^\top R_r^\top, \qquad (12)$$

where R_h, R_p, and R_r are given by

$$\begin{bmatrix} ch & sh & 0 \\ -sh & ch & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} cp & 0 & -sp \\ 0 & 1 & 0 \\ sp & 0 & cp \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & cr & sr \\ 0 & -sr & cr \end{bmatrix} \qquad (13)$$

with $c\cdot$ and $s\cdot$ shorthand for $\cos(\cdot)$ and $\sin(\cdot)$.

Position data from the USBL is sampled at a lower update rate than the IMU. For any s_k where such position data is available, we have the measurement model

$$H_{s_k} = \begin{bmatrix} I_{3\times3} & 0_{3\times6} \\ 0_{3\times6} & R(\varphi) \end{bmatrix}, \quad z_{s_k} = \begin{bmatrix} \text{usbl}^\top & \ddot{x}_\text{meas} & \ddot{y}_\text{meas} & \ddot{z}_\text{meas} \end{bmatrix}^\top$$

If there is no position data measured at time s then we use the model

$$H_{s_k} = \begin{bmatrix} 0_{3\times3} & 0_{3\times6} \\ 0_{3\times6} & R(\varphi) \end{bmatrix}, \quad z_{s_k} = \begin{bmatrix} 0 & \ddot{x}_\text{meas} & \ddot{y}_\text{meas} & \ddot{z}_\text{meas} \end{bmatrix}^\top.$$

The covariance used for measurement data similarly depends on whether there is position data available:

$$R_{s_k} = \begin{bmatrix} U_s & 0_{3\times3} \\ 0_{3\times3} & r_s I_{3\times3} \end{bmatrix}, \quad R_{s_k} = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & r_s I_{3\times3} \end{bmatrix}$$

where U is a diagonal matrix reflecting position uncertainty, while r_s captures uncertainty in IMU measurements.

Bias model. To compensate for the bias in acceleration data we augment the state vector to include bias variables:

$$\bar{x}_s = [x_s^T, b_1, b_2, b_3]^\top$$

where b_1, b_2, b_3 are bias terms for acceleration in the x, y, z directions in the local frame. To pass the bias estimates forward in time the process matrix is augmented with an identity block.

$$\bar{x}_{s_{k+1}} = \bar{F}_{s_k} x_{s_k} + \bar{w}_{s_k} \qquad (14)$$

$$\bar{F}_{s_k} = \begin{bmatrix} F_{s_k} & 0 \\ 0 & I \end{bmatrix} \qquad (15)$$

At the first time point ($s_k = 1$) we augment the covariance matrix with an identity block, and at all other time points we augment with a zero block. This adds equality constraints for the bias terms over all time points. The approach generalized easily to model piecewise-constant biases over longer periods.

$$\bar{Q}_1 = \begin{bmatrix} \Gamma^T \Gamma & 0 \\ 0 & I \end{bmatrix}, \quad \bar{Q}_{s_k} = \begin{bmatrix} \Gamma^T \Gamma & 0 \\ 0 & 0 \end{bmatrix} \quad s_k > 1.$$

The measurement matrices are augmented with an identity block that shifts the acceleration measurements using the bias. **Discretization model.** The measurement loss function is chosen to account for the level of discretization present in the data. The Vapnik loss (Figure 2f) has a 'deadzone' around the origin where small discrepancies are not penalized. This region is set using the level of discretization in the data, in this case 0.05. The sharp corners of the Vapnik loss encourage errors to lie exactly in them, an unnecessary artifact. Thus we use the 'Huberized' version dubbed 'hubnik' (Figure 2 g). The prox operators for both losses are computed in Table I.

V. ANALYSIS OF MOORING DATA

We are interested in the ability to maintain an accurate position estimate on-board an autonomous underwater vehicle in real-time using acceleration measurements from a low-cost IMU, given periodic position fixes.

To test this, we use the singular robust Kalman framework to analyze data collected from a surface mooring equipped with an IMU on the subsea node. The mooring node, which is drifting with the current, is used as a proxy for a slowly moving underwater vehicle subject to unknown disturbances. We look at the position uncertainty and error accrued over time between the periodic, world-referenced position fixes provided by the USBL system.

Data description. Position fixes are available from a ship-based Sonardyne Ranger 2 USBL every 2 seconds, which we subsample to varying degrees for the analysis. Linear acceleration data from an LSM303D 3-axis accelerometer was collected at ~ 0.075 m/s^2 precision using a Raspberry Pi Zero.
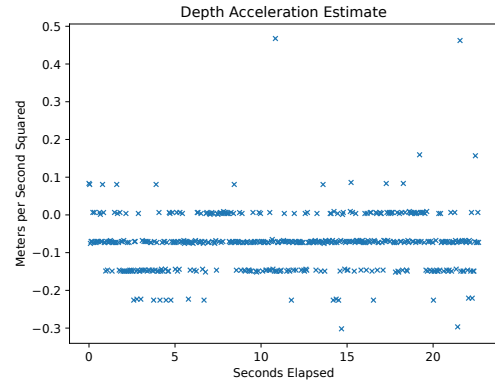

Fig. 3: A snippet of the depth acceleration data, rotated into the world frame, shows the discretization and bias of the acceleration data.

A small snippet of vertical acceleration data, Figure 3, shows the relatively coarse discretization of the measurements along with a mean that is shifted away from zero indicating bias. Outliers also appear likely. Because the IMU was generally upright with small perturbations in pitch and roll, the discretizations in the z-axis of the instrument frame are still clearly visible when the data are rotated into the vertical world frame. The small
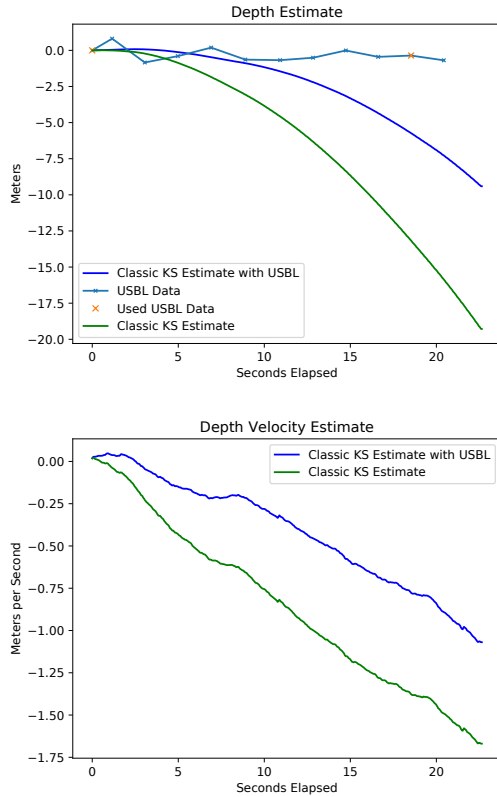
Fig. 4: Comparison of classic Kalman smoothing applied to depth data with and without USBL position fixes.

Fig. 5: Acceleration estimates of classic Kalman smoother without USBL data (top) and with USBL data (bottom).

perturbations in pitch and roll cause the slight variations visible within each discretization level.

To illustrate our model's usefulness on a data set such as in Figure 3, we isolate a 25-second window of the data and add modeling elements one at a time, noting the improvements they provide. We finish by running the full model with varying amounts of infrequent position data on a 10-minute section of data as a more practical experiment.

Model elements applied to data. We consider 25 seconds of IMU data and apply a classic Kalman filter (using least squares measurement and process loss) with the singular navigation model detailed in section IV. In practice, one would expect an underwater vehicle to be well instrumented in depth, but for illustration purposes we focus on depth and the vertical acceleration measurements. These measurements are the most biased and therefore improvements made by adding modeling elements are most clearly shown.

The 25 seconds of IMU data starts at a USBL measurement. Position is initialized to this starting USBL measurement, but for ease of comparison that position is treated as (0,0,0) and all subsequent USBL measurements are treated as relative offsets. DRS works better when its reasonably initialized. The initial state vector over the smoothing window is populated by setting acceleration to 0, and then propagating forward the most recent
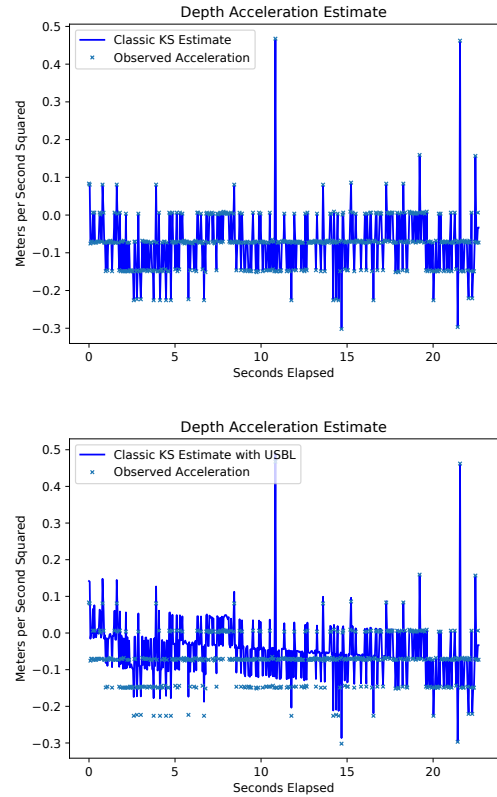
position fix with a significantly damped measurement of the most recent velocity to prevent divergence of the initial vector.

We start by adding bias estimation (for the acceleration measurements). This requires a second USBL position fix, 19 seconds into the 25-second data series. Figure 4 shows position and velocity estimates from a classic Kalman smoother applied to depth acceleration data with and without additional USBL position data. As expected due to the quality of the data we see poor performance in both cases although the small amount of additional position data offers slight improvement.

Figure 5 shows acceleration estimates of the classic Kalman smoother with and without additional USBL data. The two USBL measurements affect acceleration estimates locally but are generally overpowered by the vastly more complete set of (biased) acceleration data.

The picture improves significantly when we add in bias estimation (via singular process models), using the same two observations (compare Figures 4 and 6). With the bias removed, we can now focus removing the noticeable outliers form the acceleration data. For this purpose, we use the robust hubnik (Figure 2g) as our measurement loss. The 'deadzone' is designed to work with the coarse discretization of the acceleration data.

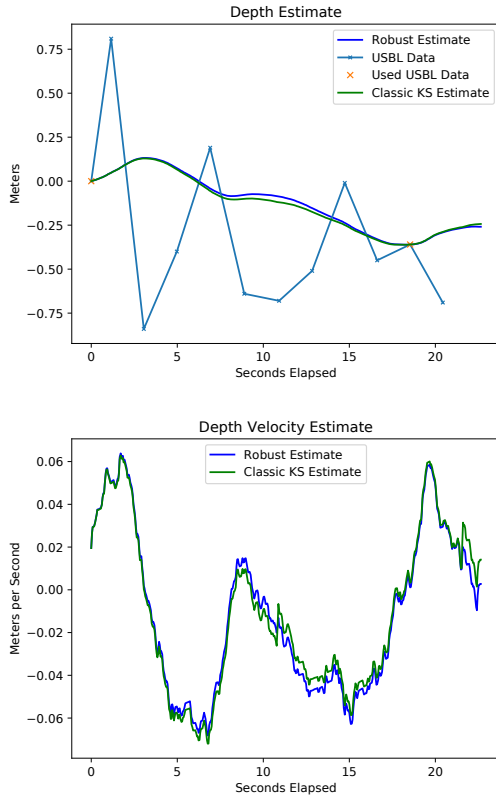Figure 6 compares the results of (debiased) fitting between

Fig. 6: Comparison (position and velocity estimates) of debiasing least squares Kalman smoother vs. a robust debiasing smoother equipped with the hubnik loss (Figure 2g).
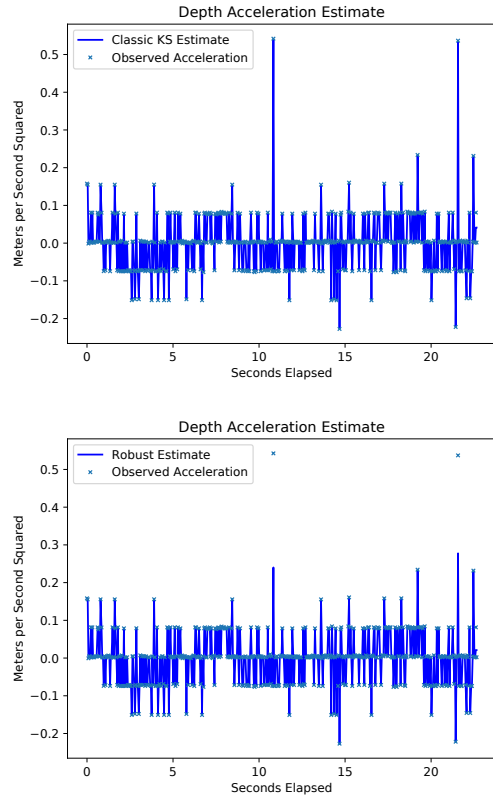

Fig. 7: Acceleration estimates for least squares debiasing Kalman smoother (top) vs. robust debiasing smoother equipped with the smoothed Vapnik loss (Figure 2g) (bottom).

the least squares Kalman smoother and the robust version using the hubnik loss. Figure 7 shows the acceleration estimates with bias removed in both the classic and robust setting. There are noticeable differences in the acceleration estimates at around 11 and 22 seconds. When using the robust smoother, the effect of the outliers on the model's estimate is greatly reduced (see the velocity estimates at 11 and 22 seconds). In the context of real-time tracking or forecasting, these sudden jumps will yield inaccurate predictions.

10 minutes of data. We apply the robust smoother with with bias, outliers, and discretization modeling elements to 10 minutes of IMU data. USBL data is available approximately every 2 seconds, but we test performance of the smoothing algorithm at larger gaps, with USBL data supplied at $30, 60$, and 120 seconds.

Figure 8 has the fitted position plots for all three frequencies. We see that without any USBL fix data, the estimate suffers; but even infrequent fixes give significant improvements when using the full capability of the smoother. In fact there are diminishing returns in increasing the USBL frequency; this is a promising result towards the future goal of a practical online implementation, particularly in settings where high-quality USBL observations are unavailable (e.g. during dives).

Figure 9 shows the fitted velocity. Here the effect of additional USBL data are more apparent, as velocity is completely inferred from position and acceleration. However, the smoothing estimates of velocity at infrequent USBL fixes are still very good compared to those informed by frequent USBL fixes.

VI. DISCUSSION AND FUTURE WORK

We have proposed a singular Kalman smoothing framework that can use singular covariance models for process and measurements, convex robust losses, and state-space constraints. The modeler can use any convex loss that has an implementable prox, a class that includes the most common choices used for inference in tracking and navigation. The framework offers a wide range of flexibility that can be used to either counteract undesirable characteristics present in data or be used to increase model performance based off of relevant field knowledge. Future work will consider real-time implementation, as well as extension to nonlinear models.

Numerical experiments show that these modeling elements yield significant improvement on a noisy, challenging dataset. We also see that having a robust model makes the smoother less reliant on frequent high-quality position updates, which is a very promising development for underwater navigation.
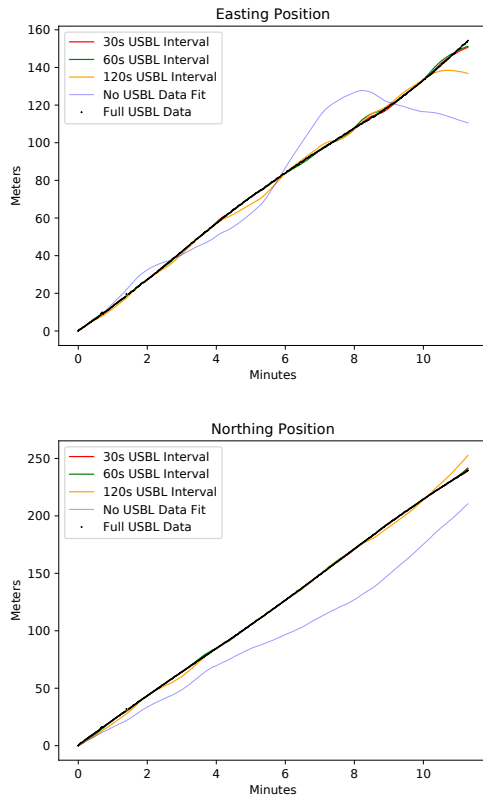
Fig. 8: Position estimates obtained with robust debiasing smoother for three frequencies of USBL fixes. Robust smoothing allows reasonable tracking from infrequent USBL observations.
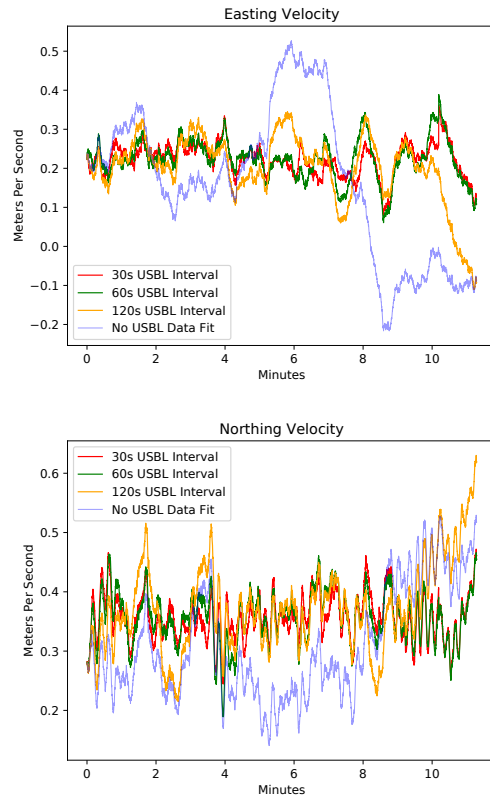

Fig. 9: Velocity estimates obtained with robust debiasing smoother for different frequencies of position data. Errors from acceleration measurements build up without USBL fixes, but infrequent USBL measurements still allow velocity estimation.

This paper develops several tools required to move to robust singular tracking in real-time. A promising aspect of singular noise models is that they make it possible to do simple robust windowed smoothing, where estimates are constrained between windows as the tracking proceeds. Constraints on the state may play a bigger role in real-time estimation, since they can help detect outliers faster. Finally, robust penalties that provide better estimates may further improve performance of the DRS algorithm, by providing an effective initialization for each new window. We will focus on these developments in future work.

Acknowledgements

References

[1] R. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[2] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Trans. ASME J. Basic Eng*, vol. 83, pp. 95–108, 1961.

[3] H. E. Rauch, C. Striebel, and F. Tung, "Maximum likelihood estimates of linear dynamic systems," *AIAA journal*, vol. 3, no. 8, pp. 1445–1450, 1965.

[4] D. Fraser and J. Potter, "The optimum linear smoother as a combination of two optimum linear filters," *IEEE Transactions on automatic control*, vol. 14, no. 4, pp. 387–390, 1969.

[5] S. A. Kassam and H. V. Poor, "Robust Techniques for Signal Processing: A Survey," *Proceedings of the IEEE*, vol. 73, no. 3, pp. 433–481, Mar. 1985.

[6] I. C. Schick and S. K. Mitter, "Robust Recursive Estimation in the Presence of Heavy-Tailed Observation Noise," *The Annals of Statistics*, vol. 22, no. 2, pp. 1045–1080, Jun. 1994.

[7] L. Fahrmeir and H. Kaufmann, "On Kalman Filtering,

Posterior Mode Estimation, and Fisher Scoring in Dynamic Exponential Family Regression," *Metrika*, pp. 37–60, 1991.

[8] B. Bell, "The Iterated Kalman Smoother as a Gauss-Newton Method," *SIAM J. Optimization*, vol. 4, no. 3, pp. 626–636, Aug. 1994.

[9] S.-J. Kim, K. Koh, S. Boyd, and D. Gorinevsky, "ℓ_1 trend filtering," *SIAM review*, vol. 51, no. 2, pp. 339–360, 2009.

[10] A. Y. Aravkin, B. M. Bell, J. V. Burke, and G. Pillonetto, "An ℓ_1-Laplace robust Kalman smoother," *IEEE Transactions on Automatic Control*, vol. 56, no. 12, pp. 2898–2911, 2011.

[11] S. Farahmand, G. B. 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.

[12] A. Y. Aravkin, J. V. Burke, and G. Pillonetto, "Sparse/robust estimation and Kalman smoothing with nonsmooth log-concave densities: Modeling, computation, and theory," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2689–2728, 2013.

[13] A. Aravkin, J. Burke, and G. Pillonetto, "Robust and trend-following student's t Kalman smoothers," *SIAM Journal on Control and Optimization*, vol. 52, no. 5, pp. 2891–2916, 2014.

[14] B. M. Bell, J. V. Burke, and G. Pillonetto, "An inequality constrained nonlinear Kalman–Bucy smoother by interior point likelihood maximization," *Automatica*, vol. 45, no. 1, pp. 25–33, 2009.

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

[16] J. Jonker, A. Aravkin, J. V. Burke, G. Pillonetto, and S. Webster, "Fast robust methods for singular state-space models," *Automatica*, vol. 105, pp. 399–405, 2019.

[17] R. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder, *Forecasting with exponential smoothing: the state space approach*. Springer Science & Business Media, 2008.

[18] C. K. Chui, G. Chen *et al.*, *Kalman filtering*. Springer, 2017.

[19] C. F. Ansley and R. Kohn, "A geometrical derivation of the fixed interval smoothing algorithm," *Biometrika*, vol. 69, no. 2, pp. 486–487, 1982.

[20] S. J. Koopman, "Exact initial Kalman filtering and smoothing for nonstationary time series models," *Journal of the American Statistical Association*, vol. 92, no. 440, pp. 1630–1638, 1997.

[21] S. Ko and R. R. Bitmead, "State estimation for linear systems with state equality constraints," *Automatica*, vol. 43, no. 8, pp. 1363–1368, 2007.

[22] B. Ait-El-Fquih and F. Desbouvries, "Fixed-interval Kalman smoothing algorithms in singular state–space systems," *Journal of Signal Processing Systems*, vol. 65, no. 3, pp. 469–478, 2011.

[23] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, no. 1-3, pp. 293–318, 1992.

[24] D. Davis and W. Yin, "Convergence rate analysis of several splitting schemes," in *Splitting Methods in Communication, Imaging, Science, and Engineering*. Springer, 2016, pp. 115–163.

[25] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*. Springer, 2011, pp. 185–212.

[26] R. T. Rockafellar and R. J.-B. Wets, *Variational analysis*. Springer Science & Business Media, 2009, vol. 317.

[27] C. Chui and G. Chen, *Kalman Filtering with Real-Time Applications*, 4th ed. Springer, 2009.

[28] D. A. Freedman, *Statistical models: theory and practice*. Cambridge University Press, 2009.

[29] G. Seber and C. Wild, *Nonlinear Regression*, ser. Wiley Series in Probability and Statistics. Wiley, 2003.

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

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

[32] R. Koenker and G. Bassett, "Regression quantiles," *Econometrica*, pp. 33–50, 1978.

[33] R. Koenker and O. Geling, "Reappraising medfly longevity: A quantile regression survival analysis," *Journal of the American Statistical Association*, vol. 96, pp. 458–468, 2001.

[34] R. A. Maronna, D. Martin, and Yohai, *Robust Statistics*, ser. Wiley Series in Probability and Statistics. Wiley, 2006.

[35] A. Aravkin, P. Kambadur, A. Lozano, and R. Luss, "Orthogonal matching pursuit for sparse quantile regression," in *Data Mining (ICDM), International Conference on*. IEEE, 2014, pp. 11–19.

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

[37] W. Chu, S. S. Keerthi, and C. J. Ong, "A unified loss function in bayesian framework for support vector regression," *Epsilon*, vol. 1, no. 1.5, p. 2, 2001.

[38] Y.-J. Lee, W.-F. Hsieh, and C.-M. Huang, "ϵ-ssvr: a smooth support vector machine for ϵ-insensitive regression," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 17, no. 5, pp. 678–685, 2005.

[39] 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.

[40] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley and Sons, 2001.