Robust State Estimation with Sparse Outliers

Citation: Graham, Matthew C., Jonathan P. How, and Donald E. Gustafson. "Robust State Estimation with Sparse Outliers." Journal of Guidance, Control, and Dynamics 38.7 (2015): 1229–1240.

As Published: http://dx.doi.org/10.2514/1.g000350

Publisher: American Institute of Aeronautics and Astronautics

Persistent URL: http://hdl.handle.net/1721.1/105810

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Robust State Estimation with Sparse Outliers

Matthew C. Graham[1] and Jonathan P. How[2]
Aerospace Controls Laboratory, MIT, Cambridge, MA, 02139

Donald E. Gustafson[3]
Draper Laboratory, Cambridge, MA, 02139

One of the major challenges for state estimation algorithms, such as the Kalman filter, is the impact of outliers that do not match the assumed Gaussian process and measurement noise. When these errors occur they can induce large state estimate errors and even filter divergence. This paper presents a robust recursive filtering algorithm, the l_1-norm filter, that can provide reliable state estimates in the presence of both measurement and state propagation outliers. The algorithm consists of a convex optimization to detect the outliers followed by a state update step based on the results of the error detection. Monte Carlo simulation results are presented to demonstrate the robustness of the l_1-norm filter estimates to both state prediction and measurement outliers. Finally, vision-aided navigation experimental results are presented that demonstrate that the proposed algorithm can provide improved state estimation performance over existing robust filtering approaches.

I. Introduction

In state estimation problems, it is often assumed that the process and measurement noise in the system are Gaussian distributed [1, 2]. However, for many practical problems the Gaussian assumption is violated by difficult to model errors (i.e. multipath [3], state prediction errors in target tracking [2]) that can be interpreted as outliers relative to the nominal Gaussian noise distribution.

[1] Draper Laboratory Fellow, Ph.D. candidate, Dept. of Aeronautics and Astronautics, MIT mcgraham@mit.edu
[2] Richard C. Maclaurin Professor of Aeronautics and Astronautics, MIT jhow@mit.edu
[3] Distinguished Member of the Technical Staff, Draper Laboratory degustafson@draper.com

Moreover, algorithms such as the Kalman filter and extended Kalman filter are not robust to outliers and the accuracy of their state estimates significantly degrades when the Gaussian noise assumption does not hold [4].

A number of robust state estimation algorithms have been developed to mitigate the impact of outliers. Typically these algorithms focus on determining when measurements are corrupted with outliers and either ignoring them entirely [5] or reducing their effect on the updated state estimates [4, 6–9]. Unfortunately, by focusing solely on measurement errors, these algorithms can not guarantee good performance when there are also large errors in the state predictions. In those cases, the algorithms incorrectly detect outliers in the measurements and end up ignoring information that could help correct the erroneous state estimates [10].

The main contribution of this paper is a robust recursive filtering algorithm, the l_1-norm filter, that can provide accurate state estimates in the presence of both state prediction and measurement outliers. The l_1-norm filter detects the presence of outliers using the solution of a convex program. Given that information, the filter updates the state estimates by jointly estimating the detected errors and the states using the information filter [11]. The algorithm is computationally efficient as it combines a convex optimization with standard recursive filtering steps.

To demonstrate the effectiveness of the l_1-norm filter, it is evaluated and compared against other robust filtering approaches using both Monte Carlo simulations and experimental data. The experimental dataset presents an urban vision-aided navigation scenario with GPS and stereo visual odometry measurements. In the context of vision-aided navigation, the state prediction errors correspond to the accumulated drift and bias errors from the visual odometry while the measurement outliers correspond to GPS errors such as multipath. These simulations and experiments demonstrate that the l_1-norm filter can match the performance of state-of-the-art robust filtering algorithms, when measurement outliers are present. More significantly, the l_1-norm filter can produce accurate state estimates in the presence of both state prediction and measurement outliers, which none of the other robust filtering algorithms can guarantee.

The paper is organized as follows. Section II discusses related work in robust filtering. The l_1-norm filter is presented in Section III. Performance evaluations of the l_1-norm filter using Monte

Carlo simulations are shown in Section IV, finally experimental results are given in Section V, and conclusions and future work are presented in Section VI.

II. Related Work

This section discusses related work in robust state estimation and provides a brief summary of previous work on the l_1-norm filter.

A. Robust State Estimation

A number of robust filtering algorithms have been developed using concepts from robust statistics [4, 6]. A major drawback of these algorithms is that they can only handle situations with measurement outliers or state propagation outliers, but not both. The H_∞ filter is an alternative robust filtering algorithm that minimizes the worst case estimation error given arbitrary noise [12–14]. Although the H_∞ filter does guarantee bounded state estimation error, under nominal noise conditions (i.e. white, zero-mean and Gaussian), the H_∞ filter estimates can in fact be worse (in a least-squares sense) than those generated by the Kalman filter (KF) [14]. The filter presented in this paper avoids the downsides of both the robust statistics-based and H_∞ filters because it can provide robust state estimates in the presence of simultaneous state propagation and measurement outliers. Furthermore, it is shown in this paper that under nominal (i.e. Gaussian) noise conditions the l_1-norm filter solution is equivalent to the KF solution.

More recently, a number of robust filtering algorithms have been developed that adapt the system noise parameters over time using variational Bayesian (VB) inference [7–9, 15]. VB filters achieve robust estimates by introducing uncertainty into the system noise model (i.e. assuming that the mean and covariance of the noise are random variables) and then jointly solving for the states and noise parameters at each time step. This procedure provides robustness to unmodeled measurement errors because the update procedure can adjust the measurement noise covariance online and as a result can increase the covariance of corrupted measurements. This in turn reduces the impact of the measurement errors on the state estimates because the gain for the measurements is decreased as their noise covariance increases. While the state and parameter updates are iterative for these methods, they closely resemble the KF updates and usually exhibit fast convergence [7, 15]. The

major drawback of these methods is that they do not try to adapt the process noise parameters, and as a result may be susceptible to unmodeled errors in the state propagation. This behavior is demonstrated both in simulation and experimentally in Sections IV and V of this paper.

The l_1-norm filter is similar to convex optimization based robust filtering algorithms proposed by Mattingley and Boyd [16] and Kim et al. [17]. However, a major difference between the algorithms is the set of models that are used. The models used in the other convex optimization based filters only consider measurement outliers. As with the other robust filtering algorithms discussed, by ignoring the possibility of large state prediction errors, the filters proposed by Mattingley and Boyd, and Kim et al. can not guarantee robust state estimates when they occur. In contrast, the l_1-norm filter considers a more general set of error models in the system dynamics, i.e. both state prediction and measurement outliers, that reduce to those used by Mattingley and Boyd, and Kim et al. if the state prediction errors are zero.

All of the convex optimization based algorithms solve an l_1-norm minimization problem to calculate state estimates that are robust to unmodeled errors. The l_1-norm filter differs from the other convex optimization approaches to robust filtering because it does not use the output of the l_1-norm minimization directly in the state update equations, which can lead to several issues with the final state estimates. First, the error estimates generated by the l_1-norm minimization are biased [18], which can in turn cause the state estimates to be biased. Additionally, there is no clear way to calculate the covariance of the error estimates using the l_1-norm minimization. This means that there is no way to account for the correlations between the error estimates and the state estimates, which can impact the accuracy of the state covariance calculations. The robust filter developed by Mattingley and Boyd assumes that the filter covariance has reached steady-state so that the covariance of the error estimates is unnecessary. None of the other convex optimization filters address the issue of bias induced by the l_1-optimal solution. In contrast, the l_1-norm filter presented here provides an unbiased estimate and a proper accounting of the state covariance by jointly estimating the state and the non-zero error terms detected by the l_1-norm minimization using the information filter.

There have also been several recent Kalman filtering techniques proposed in the compressed

sensing literature (CS-KF approaches) that contain an l_1-norm minimization as a subroutine [19–21]. The CS-KF algorithms are used to estimate *sparse state vectors* and apply an l_1-norm minimization to promote sparsity in the state estimates. In contrast, the l_1-norm filter estimates a *dense state vector* and applies the l_1-minimization as a means of detecting sparse outliers that have corrupted the measurements and state predictions. Thus, while both approaches apply similar algorithmic techniques the problems they are solving are quite different.

B. Previous Work on the l_1-norm filter

This paper builds on and extends previous work on the l_1-norm filter that was presented by Mohiuddin et al. [22]. This paper presents a formal derivation of the l_1-norm filter update equations that was not discussed in the prior publications. This analysis provides additional insight into the l_1-norm filter update equations and how they relate to standard state estimation algorithms like the Kalman filter. Finally, this paper also analyzes the impact of incorrect outlier detection on the filter estimates which had not been considered or investigated in previous work.

III. Robust State Estimation Using the l_1-norm Filter

This section develops the l_1-norm filter algorithm which consists of two parts:

1. Identification of outliers in the state propagation and measurements by solving a convex optimization problem
2. Updating the state estimates given the results of the error identification step

A. System Models and Problem Statement

It will be assumed that the state dynamics and measurements are linear and corrupted by both additive white Gaussian noise as well as additive sparse errors. Sparse in this context means that at least some components of the errors are equal to zero. Given these assumptions, the state propagation and measurement models are assumed to take the form:

$$\mathbf{x}_{k+1} = F_k \mathbf{x}_k + \mathbf{w}_k + \mathbf{e}_k^p \tag{1}$$

$$\mathbf{y}_{k+1} = H_{k+1} \mathbf{x}_{k+1} + \mathbf{v}_{k+1} + \mathbf{e}_{k+1}^m \tag{2}$$

where F_k is the state transition matrix, H_{k+1} is the measurement matrix, $\mathbf{w}_k$ and $\mathbf{v}_{k+1}$ are the Gaussian process and measurement noise, respectively, and $\mathbf{e}_k^p$ and $\mathbf{e}_{k+1}^m$ represent the sparse errors. Note that without the errors, $\mathbf{e}_k^p$ and $\mathbf{e}_{k+1}^m$, these equations are in the standard form of the KF state propagation and measurement equations. The rest of the assumptions for the state prediction and measurement models are as follows:

1. $\mathbf{w}_k$ and $\mathbf{v}_{k+1}$ are white and zero-mean with covariances Q_k and R_{k+1} respectively
2. $\mathbf{w}_k$ and $\mathbf{v}_{k+1}$ are mutually uncorrelated (i.e. $E[\mathbf{v}_{k+1}\mathbf{w}_k^T] = 0, \forall\ k$)
3. The number of combined non-zero components of $\mathbf{e}_k^p$ and $\mathbf{e}_{k+1}^m$ is less than or equal to the number of measurements

The first two assumptions are standard for the KF. The final assumption about the sparse errors is required to ensure a valid state estimate using the l_1-norm filter. The third assumption is discussed in more detail during the state-update portion of this section.

The objective of the state estimation problem is to calculate a state estimate, $\hat{\mathbf{x}}_{k+1|k+1}$, that minimizes the mean squared state estimation error, $E[(\mathbf{x}_{k+1|k+1} - \hat{\mathbf{x}}_{k+1|k+1})^T(\mathbf{x}_{k+1|k+1} - \hat{\mathbf{x}}_{k+1|k+1})]$ given an estimate of the state at time k, $\hat{\mathbf{x}}_{k|k}$, and a set of measurements up to time $k+1$. It will be assumed that the estimation error at time k, $\tilde{\mathbf{x}}_{k|k} = \mathbf{x}_k - \hat{\mathbf{x}}_{k|k}$, is zero-mean and Gaussian distributed with covariance $P_{k|k}$.

For a system with state dynamics and measurements governed by (1) and (2), solving for $\hat{\mathbf{x}}_{k+1|k+1}$ also entails solving for $\hat{\mathbf{e}}_{k|k+1}^p$ and $\hat{\mathbf{e}}_{k+1|k+1}^m$. It should be noted that without the sparsity assumption, this estimation problem is potentially ill-posed and could have multiple solutions. The sparsity assumption acts as a regularizer for the estimation problem that favors "simpler" explanations of the measurements when outliers occur.

B. Error Detection

This section formulates an optimization problem that can approximately solve for the sparse errors. The output of the optimization will be used to determine which components of $\mathbf{e}_k^p$ and $\mathbf{e}_{k+1}^m$ are non-zero. Discussion of how the states and errors are estimated given this information will be

covered in the next subsection.

Before the optimization problem can be defined, the measurement residuals need to be expressed in terms of the sparse errors, $\mathbf{e}_k^p$ and $\mathbf{e}_{k+1}^m$, and the *a priori* state estimate. The *a priori* measurement residuals at time $k+1$ can be expressed as:

$$\begin{aligned}\tilde{\mathbf{y}}_{k+1} &= \mathbf{y}_{k+1} - H_{k+1}F_k\hat{\mathbf{x}}_{k|k} \\ &= H_{k+1}\left(F_k\mathbf{x}_k + \mathbf{w}_k + \mathbf{e}_k^p - \hat{\mathbf{x}}_{k+1|k}\right) + \mathbf{v}_{k+1} + \mathbf{e}_{k+1}^m \quad (3) \\ &= H_{k+1}\left(F_k\tilde{\mathbf{x}}_{k|k} + \mathbf{w}_k + \mathbf{e}_k^p\right) + \mathbf{v}_{k+1} + \mathbf{e}_{k+1}^m \quad (4)\end{aligned}$$

After rearranging terms in (4) and defining $\mathbf{e}_{k+1} \equiv \begin{bmatrix} \mathbf{e}_k^p \\ \\ \mathbf{e}_{k+1}^m \end{bmatrix}$ and $\mathbf{u}_{k+1} \equiv H_{k+1}\left(F_k\tilde{\mathbf{x}}_{k|k} + \mathbf{w}_k\right) + \mathbf{v}_{k+1}$, the residuals can be related to the error terms by

$$\tilde{\mathbf{y}}_{k+1} = \begin{bmatrix} H_{k+1} & I \end{bmatrix}\mathbf{e}_{k+1} + \mathbf{u}_{k+1} \quad (5)$$

The errors could be estimated from the under-determined system of equations in (5) by solving for the minimum l_2-norm vector that corresponds to the measurement residuals (using a pseudo-inverse least squares solution [11]). However, this approach is not suitable for estimating sparse vectors such as $\mathbf{e}_{k+1}$ because it tends to allocate signal energy to all of the components of the vector being estimated instead of concentrating it on a few components, thus returning a non-sparse estimate of a sparse vector.

Based on the sparsity assumption, the estimates for $\mathbf{e}_{k+1}$ should have as few non-zero entries as possible. Additionally, if the error estimates are equal to the true error values (i.e. $\hat{\mathbf{e}}_{k+1} = \mathbf{e}_{k+1}$) then the corrected measurement residuals, $\tilde{\mathbf{y}} = \tilde{\mathbf{y}}_{k+1} - \begin{bmatrix} H_{k+1} & I \end{bmatrix}\hat{\mathbf{e}}_{k+1}$, will be equal to $\mathbf{u}_{k+1}$. Note that $\mathbf{u}_{k+1}$ is a zero-mean normally distributed random variable with covariance

$$\Sigma = H_{k+1}\left(F_kP_{k|k}F_k^T + Q_k\right)H_{k+1}^T + R_{k+1}$$

For a normally distributed random variable $\mathbf{p} \in \mathcal{R}^n$ with covariance, W, the weighted inner product $\mathbf{p}^TW^{-1}\mathbf{p}$ is χ^2 distributed with n degrees of freedom. Given these observations, one way to obtain a good estimate of $\mathbf{e}_{k+1}$ is to minimize the number of non-zero entries while ensuring that $\tilde{\mathbf{y}}^T\Sigma^{-1}\tilde{\mathbf{y}} \leq \tau$, where τ is set based on the χ^2 c.d.f. [18]. Mathematically this optimization can be

expressed as

$$\min_{\hat{\mathbf{e}}_{k+1}} \quad \|\hat{\mathbf{e}}_{k+1}\|_0 \tag{6}$$
$$\text{subject to} \quad \tilde{\mathbf{y}}^T \Sigma^{-1} \tilde{\mathbf{y}} \leq \tau$$

where $\|\cdot\|_0$ is a shorthand expression for the number of non-zero components of a vector [23]. Because this optimization involves searching over a combinatorial set of sparse vectors, it is computationally intractable in general [23]. Fortunately, a tractable approximate solution to (6), can be found by solving the convex optimization [18]

$$\min_{\hat{\mathbf{e}}_{k+1}} \quad \|\hat{\mathbf{e}}_{k+1}\|_1 \tag{7}$$
$$\text{subject to} \quad \tilde{\mathbf{y}}^T \Sigma^{-1} \tilde{\mathbf{y}} \leq \tau$$

The optimization in (7) can be recast as a second-order cone program for which a number of efficient algorithms have been developed [24, 25].

In practice, the optimization posed in (7) is acting as a consistency check between the measurements and the *a priori* state estimate generated by the nominal state propagation model, $\hat{\mathbf{x}}_{k+1} = F_k \hat{\mathbf{x}}_k$. If there is an inconsistency, then the l_1 minimization can both detect and attribute it to specific error sources in the measurements and state propagation in one computationally efficient step. In the case where no errors are present, then the residuals should already satisfy the inequality constraint and the error estimates will be equal to zero.

Although the l_1-minimization step tends to return a sparse estimate of the errors, the estimate often has small spurious non-zero components that are a result of measurement noise. To ensure that the error estimates are sufficiently sparse, the solution returned by the l_1-minimization is thresholded based on the expected noise level. Any elements of the l_1-optimal error estimates that are smaller than the expected noise level (as determined by a χ^2-test) are set to zero. This step ensures that only errors that are inconsistent with the Gaussian process and measurement noise are considered in the state update update portion of the algorithm. Sparse estimates of the errors could also be obtained by applying the reweighted l_1-norm minimization (RWL1) approach proposed by Candes et al. [26]. However, since RWL1 requires iteratively solving an l_1-minimization multiple

times, it remains to be seen, if the solution can be generated at the high rate needed for navigation systems.

It should also be noted that while there is extensive evidence in the compressed sensing literature that the l_1-norm minimization encourages sparse solutions [26] the solution to Equation 7 is not guaranteed to coincide with the solution to Equation 6. The impact of missed detections and false alarms in the error detection procedure will be discussed in more detail in Section III E.

C. State and Error Estimation

After performing the error detection, the state estimates are updated by augmenting the state vector with the non-zero error terms and then jointly estimating the errors and states using the information filter. The combination of thresholding the l_1-optimal solution followed by re-estimation is a common procedure in sparse signal estimation usually referred to as debiasing [18, 27], because in practice the l_1-optimal solutions are biased [18].

The information filter is a recursive filter that is algebraically equivalent to the KF [28], but performs operations on the information matrix, $\Lambda_{k|k}$, and information state, $\hat{\mathbf{d}}_{k|k}$ instead of the state and covariance. Given a state estimate, $\hat{\mathbf{x}}_{k|k}$, and covariance, $P_{k|k}$, the information matrix and state are defined as:

$$\Lambda_{k|k} = \left(P_{k|k}\right)^{-1} \tag{8}$$

$$\hat{\mathbf{d}}_{k|k} = \Lambda_{k|k}\hat{\mathbf{x}}_{k|k} \tag{9}$$

The information filter is particularly useful for situations where some of the states have uninformative prior estimates (such as the non-zero terms of $\mathbf{e}_k^p$ and $\mathbf{e}_{k+1}^m$).

The *a priori* measurement residuals in (3) will be used to derive the information filter update for the state and error estimates. First, define the augmented state vector $\mathbf{z}_{k+1}$ as

$$\mathbf{z}_{k+1} = \begin{bmatrix} \bar{\mathbf{x}}_{k+1} \\ \mathbf{e}_k^{p,nz} \\ \mathbf{e}_{k+1}^{m,nz} \end{bmatrix} \tag{10}$$

where $\bar{\mathbf{x}}_{k+1} = F_k\mathbf{x}_k + \mathbf{w}_k$ and the superscript nz denotes only the non-zero components (as determined by the l_1-norm minimization) of the respective errors. After substituting in the definition of

$\mathbf{z}_{k+1}$, the measurements can be expressed as

$$\begin{aligned}\mathbf{y}_{k+1} &= [H_{k+1} \;\; H_p \;\; I_m]\,\mathbf{z}_{k+1} + \mathbf{v}_{k+1} \\ &= \bar{H}_{k+1}\mathbf{z}_{k+1} + \mathbf{v}_{k+1}\end{aligned} \tag{11}$$

where H_p is equal to the columns of H_{k+1} corresponding to the non-zero terms in $\hat{\mathbf{e}}_k^p$ and I_m is equal to the columns of the identity matrix corresponding to non-zero entries in $\hat{\mathbf{e}}_{k+1}^m$.

The prior estimate of $\bar{\mathbf{x}}_{k+1|k}$ can be expressed as

$$\hat{\bar{\mathbf{x}}}_{k+1|k} = F_k\hat{\mathbf{x}}_{k|k}$$

and the associated covariance is $P_{\bar{\mathbf{x}}} = F_k P_{k|k} F_k^T + Q_k$. Since the prior estimates of the errors are assumed to be uninformative, the information matrix $\hat{\mathbf{z}}_{k+1|k}$ will be

$$\Lambda_{k+1|k} = \begin{bmatrix} P_{\bar{\mathbf{x}}}^{-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{12}$$

with the information state, $\hat{\mathbf{d}}_{k+1|k}$ given by (9). After calculating the information matrix and state, they can be updated as follows [11]

$$\hat{\mathbf{d}}_{k+1|k+1} = \hat{\mathbf{d}}_{k+1|k} + \bar{H}_{k+1}^T R_{k+1}^{-1}\mathbf{y}_{k+1} \tag{13}$$

$$\Lambda_{k+1|k+1} = \Lambda_{k+1|k} + \bar{H}_{k+1}^T R_{k+1}^{-1}\bar{H}_{k+1} \tag{14}$$

After updating $\hat{\mathbf{d}}_{k+1|k+1}$ and $\Lambda_{k+1|k+1}$, the covariance $P_{k+1|k+1}^z$ and state estimate $\hat{\mathbf{z}}_{k+1|k+1}$ can be calculated from (8) and (9), respectively.

Recall that the total number of non-zero entries in $\mathbf{e}_k^p$ and $\mathbf{e}_{k+1}^m$ was assumed to be less than or equal to the number of measurements. The update procedure in (13) and (14) sets the upper bound on the allowable sparsity of the unmodeled errors. Note that the number of combined non-zero components of $\mathbf{e}_{k+1}^m$ and $\mathbf{e}_k^p$ must be less than or equal to the number of measurements in order to ensure that $\Lambda_{k+1|k+1}$ is full rank and can be inverted. If $\Lambda_{k+1|k+1}$ is singular then it can not be inverted and $\hat{\mathbf{z}}_{k+1|k+1}$ can not be calculated.

After calculating $\hat{\mathbf{z}}_{k+1|k+1}$, the posterior state estimate, $\hat{\mathbf{x}}_{k+1|k+1}$, corrected for the sparse errors,

is

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\bar{\mathbf{x}}}_{k+1|k+1} + \hat{\mathbf{e}}^p_{k|k+1} \tag{15}$$

with covariance

$$P_{k+1|k+1} = P^{\bar{\mathbf{x}}}_{k+1|k+1} + P^{\mathbf{e}^p}_{k+1|k+1} + P_{\bar{x}e} + P_{e\bar{x}} \tag{16}$$

where $P^{\bar{\mathbf{x}}}_{k+1|k+1}$ is the covariance of $\hat{\bar{\mathbf{x}}}_{k+1|k+1}$, $P^{\mathbf{e}^p}_{k+1|k+1}$ is the covariance of $\hat{\mathbf{e}}^p_{k|k+1}$, and $P_{\bar{x}e}$ and $P_{e\bar{x}}$ are the cross covariance matrices of $\hat{\bar{\mathbf{x}}}_{k+1|k+1}$ and $\hat{\mathbf{e}}^p_{k|k+1}$, all of which can be obtained from $P^z_{k+1|k+1}$:

$$P^z_{k+1|k+1} = \begin{bmatrix} P^{\bar{\mathbf{x}}}_{k+1|k+1} & P_{\bar{x}e} & \cdot \\ P_{e\bar{x}} & P^{\mathbf{e}^p}_{k+1|k+1} & \cdot \\ \cdot & \cdot & \cdot \end{bmatrix}$$

D. Algorithm Summary

The l_1-norm filter is summarized in Algorithm 1. There are two main components of the algorithm: outlier detection and a state update based on the outlier detection. Steps 1 and 2 encompass the outlier detection portion of the algorithm, where a constrained l_1-norm optimization is used to estimate the sparse vectors $\mathbf{e}^p_k$ and $\mathbf{e}^m_{k+1}$. A χ^2-test is applied to the error estimates calculated by the l_1-norm optimization to determine which non-zero components of $\hat{\mathbf{e}}^p_k$ or $\hat{\mathbf{e}}^m_{k+1}$ are too large to be explained by the Gaussian process and measurement noise. The large non-zero components of $\hat{\mathbf{e}}^p_k$ and $\hat{\mathbf{e}}^m_{k+1}$ are then re-estimated in the state update step to calculate the robust state estimates.

The state update phase of the algorithm occurs in steps 3–6. The states and non-zero components of the errors are solved for by augmenting the state vector and then processing the measurements using the information filter. This portion of the algorithm is closely related to the KF. Calculating the information state and information matrix (step 3) requires applying the KF state propagation equations, while the state update equations for the KF are analogous to steps 4–6.

Algorithm 1 l_1-norm Filter

Require: $\hat{\mathbf{x}}_{k|k}$, $P_{k|k}$, $\mathbf{y}_{k+1}$

1. Solve l_1 minimization problem in (7) for $\hat{\mathbf{e}}_{k+1}$
2. Apply χ^2-test to determine non-zero components of $\hat{\mathbf{e}}_{k+1}$
3. Form information state ($\hat{\mathbf{d}}_{k+1|k}$) and matrix ($\Lambda_{k+1|k}$) for augmented state vector $\hat{\mathbf{z}}_{k+1|k}$
4. Update $\hat{\mathbf{d}}_{k+1|k}$, $\Lambda_{k+1|k}$ with (13)–(14)
5. Calculate $P^z_{k+1|k+1}$, $\hat{\mathbf{z}}_{k+1|k+1}$ with (8)–(9)
6. Calculate $\hat{\mathbf{x}}_{k+1|k+1}$, $P_{k+1|k+1}$ using (15)–(16)

return $\hat{\mathbf{x}}_{k+1|k+1}$, $P_{k+1|k+1}$

E. Algorithm Analysis

This section will derive closed form expressions for the posterior state estimates and covariance using the l_1-norm filter. These expressions will provide additional insight into the l_1-norm filter and allow analysis of the l_1-norm filter when the errors detected by the l_1-norm minimization are incorrect.

To simplify the derivations that follow, it will be assumed without loss of generality that the states and measurements have been ordered so that they can be partitioned into subsets that are impacted by $\mathbf{e}^p_k$ and $\mathbf{e}^m_{k+1}$. After ordering the states and measurements, $\mathbf{x}_{k+1}$, $\mathbf{y}_{k+1}$, H_{k+1} and R_{k+1} can be partitioned as

$$\mathbf{x}_{k+1} = \begin{bmatrix} \mathbf{x}_{\bar{p}} \\ \mathbf{x}_p \end{bmatrix}, \quad \mathbf{y}_{k+1} = \begin{bmatrix} \mathbf{y}_u \\ \mathbf{y}_c \end{bmatrix}, \quad H_{k+1} = \begin{bmatrix} H_{\bar{p}u} & H_{pu} \\ H_{\bar{p}c} & H_{pc} \end{bmatrix}, \quad R_{k+1} = \begin{bmatrix} R_u & \mathbf{0} \\ \mathbf{0} & R_c \end{bmatrix}$$

where the subscripts u and c denoted corrupted and uncorrupted measurements respectively, and the subscripts $\bar{p}$ and p indicate state variables that are uncorrupted and corrupted by $\mathbf{e}^p_k$ respectively. Similarly, the *a priori* state information matrix and covariance matrix can be partitioned as

$$\Lambda_{k+1|k} = \begin{bmatrix} \Lambda_{\bar{p}} & \Lambda_{\bar{p}p} \\ \Lambda_{p\bar{p}} & \Lambda_p \end{bmatrix} \text{ and } P_{k+1|k} = \begin{bmatrix} P_{\bar{p}} & P_{\bar{p}p} \\ P_{p\bar{p}} & P_p \end{bmatrix}$$

The posterior covariance of the states and errors can be calculated by inverting the posterior

information matrix in (14):

$$P^z_{k+1|k+1} = \begin{bmatrix} P^{-1}_{k+1|k} + H^T R^{-1} H & H^T R^{-1} H_p & H^T R^{-1} I_m \\ H_p^T R^{-1} H & H_p^T R^{-1} H_p & H_p^T R^{-1} I_m \\ I_m^T R^{-1} H & I_m^T R^{-1} H_p & I_m^T R^{-1} I_m \end{bmatrix}^{-1}$$

Note that the general form for a blockwise inverse of a matrix is

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} (A - BD^{-1}C)^{-1} & -(A - BD^{-1}C)^{-1}BD^{-1} \\ -D^{-1}C(A - BD^{-1}C)^{-1} & D^{-1} + D^{-1}C(A - BD^{-1}C)^{-1}BD^{-1} \end{bmatrix} \tag{17}$$

The derivation of the update formulas will proceed by applying blockwise inversion using $A = P^{-1}_{k+1|k} + H^T R^{-1} H$. After selecting A, D^{-1} can also be calculated by blockwise inversion:

$$D^{-1} = \begin{bmatrix} (H_{pu}^T R_u^{-1} H_{pu})^{-1} & -(H_{pu}^T R_u^{-1} H_{pu})^{-1} H_{pc}^T \\ H_{pc}(H_{pu}^T R_u^{-1} H_{pu})^{-1} & R_c + H_{pc}(H_{pu}^T R_u^{-1} H_{pu})^{-1} H_{pc}^T \end{bmatrix} \tag{18}$$

Given these definitions, it can be shown that the covariance term $P^{\bar{x}}_{k+1|k+1}$ in (16) can be described using an update formula similar to a KF update:

Lemma 1. *$P^{\bar{x}}_{k+1|k+1} = (I - \overline{KH}) P^{\bar{x}}_{k+1|k}$ where*

$$\overline{KH} = \begin{bmatrix} \bar{K} H_{\bar{p}u} & \mathbf{0} \\ -\Lambda_p^{-1} \Lambda_{p\bar{p}} \bar{K} H_{\bar{p}u} & \mathbf{0} \end{bmatrix}, \quad \bar{K} = P_{\bar{p}} H_{\bar{p}u}^T \left(\bar{R}_u + H_{\bar{p}u} P_{\bar{p}} H_{\bar{p}u}^T \right)^{-1}$$

$$\bar{R}_u = (R_u^{-1} - R_u^{-1} H_{pu} (H_{pu}^T R_u^{-1} H_{pu})^{-1} H_{pu}^T R_u^{-1})^{-1}$$

Proof. Note that $P^{\bar{x}}_{k+1|k+1}$ is equivalent to the the top left entry in $P^z_{k+1|k+1}$. Therefore it can be expressed as

$$\begin{aligned} P^{\bar{x}}_{k+1|k+1} &= (P^{-1}_{k+1|k} + H^T R^{-1} H - BD^{-1}C)^{-1} \\ &= \begin{bmatrix} \Lambda_{\bar{p}} + H_{\bar{p}u}^T \bar{R}_u^{-1} H_{\bar{p}u} & \Lambda_{\bar{p}p} \\ \Lambda_{p\bar{p}} & \Lambda_p \end{bmatrix}^{-1} \end{aligned}$$

$P^{\bar{x}}_{k+1|k+1}$ can now be calculated using blockwise inversion, but first note that $P_{\bar{p}} = (\Lambda_{\bar{p}} - \Lambda_{\bar{p}p} \Lambda_p^{-1} \Lambda_{p\bar{p}})^{-1}$. Using this fact, the upper left hand term of $P^{\bar{x}}_{k+1|k+1}$ is

$$P^{\bar{x}}_{\bar{p}} = (\Lambda_{\bar{p}} - \Lambda_{\bar{p}p} \Lambda_p^{-1} \Lambda_{p\bar{p}} + H_{\bar{p}u}^T \bar{R}_u^{-1} H_{\bar{p}u})^{-1}$$

$$
\begin{aligned}
&= (P_{\bar{p}} + H_{\bar{p}u}^T \bar{R}_u^{-1} H_{\bar{p}u})^{-1} \\
&= P_{\bar{p}} + P_{\bar{p}} H_{\bar{p}u}^T \left(\bar{R}_u + H_{\bar{p}u} P_{\bar{p}} H_{\bar{p}u}^T\right)^{-1} P_{\bar{p}} \\
&= (I - \bar{K} H_{\bar{p}u}) P_{\bar{p}}
\end{aligned}
$$

where the third equality follows from the matrix inversion lemma.

Applying the rest of the blockwise inverse formula leads to

$$
\begin{aligned}
P_{k+1|k+1}^{\bar{x}} &= \begin{bmatrix} (I - \bar{K} H_{\bar{p}u}) P_{\bar{p}} & -(I - \bar{K} H_{\bar{p}u}) P_{\bar{p}} \Lambda_{\bar{p}p} \Lambda_p^{-1} \\ -\Lambda_p^{-1} \Lambda_{p\bar{p}} (I - \bar{K} H_{\bar{p}u}) P_{\bar{p}} & \Lambda_p^{-1} + \Lambda_p^{-1} \Lambda_{p\bar{p}} (I - \bar{K} H_{\bar{p}u}) P_{\bar{p}} \Lambda_{\bar{p}p} \Lambda_p^{-1} \end{bmatrix} \\
&= \begin{bmatrix} (I - \bar{K} H_{\bar{p}u}) P_{\bar{p}} & P_{\bar{p}p} - \bar{K} H_{\bar{p}u} P_{\bar{p}p} \\ P_{p\bar{p}} + \Lambda_p^{-1} \Lambda_{p\bar{p}} \bar{K} H_{\bar{p}u} P_{\bar{p}} & P_p + \Lambda_p^{-1} \Lambda_{p\bar{p}} \bar{K} H_{\bar{p}u} P_{\bar{p}p} \end{bmatrix} \\
&= (I - \bar{K} H) P_{k+1|k}^{\bar{x}}
\end{aligned}
$$

□

Given (Equation 18) and Lemma 1 the final form of the state estimates and covariance in the l_1-norm filter are given by the following theorem.

Theorem 1. *The l_1-norm filter state estimates in (15) can be expressed as:*

$$
\hat{\mathbf{x}}_{k+1|k+1} = \begin{bmatrix} \hat{\mathbf{x}}_{k+1|k}^{\bar{p}} + \bar{K}(\mathbf{y}_u - H_{\bar{p}u} \hat{\mathbf{x}}_{k+1|k}^{\bar{p}}) \\ H_{pu}^{-L}(\mathbf{y}_u - H_{\bar{p}u} \hat{\mathbf{x}}_{k+1|k+1}^{\bar{p}}) \end{bmatrix} \tag{19}
$$

In addition, the posterior covariance matrix in (16) is given by:

$$
P_{k+1|k+1} = \begin{bmatrix} P_{\bar{p}}^{\bar{x}} & -P_{\bar{p}}^{\bar{x}} (H_{pu}^{-L} H_{\bar{p}u})^T \\ -H_{pu}^{-L} H_{\bar{p}u} P_{\bar{p}}^{\bar{x}} & (H_{pu}^T R_u^{-1} H_{pu})^{-1} + H_{pu}^{-L} H_{\bar{p}u} P_{\bar{p}}^{\bar{x}} (H_{pu}^{-L} H_{\bar{p}u})^T \end{bmatrix} \tag{20}
$$

where

$$
H_{pu}^{-L} = \left(H_{pu}^T R_u^{-1} H_{pu}\right)^{-1} H_{pu}^T R_u^{-1}
$$

Proof. Using (9), (13) and the blockwise inversion formula for $P_{k+1|k+1}^z$, the updated state and error estimates are

$$
\hat{\tilde{\mathbf{x}}}_{k+1|k+1} = P_{k+1|k+1}^{\bar{x}} \left(P_{k+1|k}^{-1} \hat{\mathbf{x}}_{k+1|k} + H^T R^{-1} \mathbf{y} - B D^{-1} \mathbf{i}_e\right)
$$

$$\begin{aligned}\hat{\mathbf{e}}^{nz}_{k+1|k+1} &= -D^{-1}CP^{\bar{x}}_{k+1|k+1}\left(P^{-1}_{k+1|k}\hat{\mathbf{x}}_{k+1|k} + H^TR^{-1}\mathbf{y} - BD^{-1}\mathbf{i}_e\right) + D^{-1}\mathbf{i}_e \\ &= -D^{-1}C\hat{\bar{\mathbf{x}}}_{k+1|k+1} + D^{-1}\mathbf{i}_e\end{aligned}$$

where

$$\mathbf{i}_e = \begin{bmatrix} H_p^TR^{-1}\mathbf{y} \\ I_m^TR^{-1}\mathbf{y} \end{bmatrix}$$

The matrices BD^{-1} and $D^{-1}C$ arise from the blockwise inversion of $\Lambda_{k+1|k+1}$ and can be shown to be

$$BD^{-1} = \begin{bmatrix} (H^{-L}_{pu}H_{\bar{p}u})^T & H^T_{\bar{p}c} - (H^{-L}_{pu}H_{\bar{p}u})^TH^T_{pc} \\ I & \mathbf{0} \end{bmatrix}, \quad D^{-1}C = \begin{bmatrix} H^{-L}_{pu}H_{\bar{p}u} & I \\ H_{\bar{p}c} - H_{pc}H^{-L}_{pu}H_{\bar{p}u} & \mathbf{0} \end{bmatrix}$$

Note that only $\hat{\mathbf{e}}^p$ appears in (15), thus it is only necessary to calcuate $\hat{\mathbf{e}}^{p,nz}$. After substituting the values of BD^{-1}, $D^{-1}C$ and $\mathbf{i}_e$ the estimates of the non-zero $\mathbf{e}^p$ terms are:

$$\hat{\mathbf{e}}^{p,nz} = H^{-L}_{pu}(\mathbf{y}_u - H_{\bar{p}u}\hat{\bar{\mathbf{x}}}^{\bar{p}}_{k+1|k+1}) - \hat{\mathbf{x}}^p_{k+1|k+1} \tag{21}$$

After substituting BD^{-1} and the value for $P^{\bar{x}}_{k+1|k+1}$ from Lemma 1 into the update equation for $\hat{\bar{\mathbf{x}}}_{k+1|k+1}$ the result is

$$\hat{\bar{\mathbf{x}}}_{k+1|k+1} = \begin{bmatrix} \hat{\bar{\mathbf{x}}}^{\bar{p}}_{k+1|k+1} \\ \hat{\mathbf{x}}^p_{k+1|k+1} \end{bmatrix} = \begin{bmatrix} (I - \bar{K}H_{\bar{p}u})\hat{\bar{\mathbf{x}}}^p_{k+1|k} + (I - \bar{K}H_{\bar{p}u})P_{\bar{p}}H^T_{\bar{p}u}\bar{R}^{-1}_u\mathbf{y}_u \\ \hat{\mathbf{x}}^p_{k+1|k+1} \end{bmatrix} \tag{22}$$

The second term in the $\hat{\bar{\mathbf{x}}}^{\bar{p}}_{k+1|k+1}$ can be simplified using a Schur identity [29] as

$$\begin{aligned}(I - \bar{K}H_{\bar{p}u})P_{\bar{p}}H^T_{\bar{p}u}\bar{R}^{-1}_u\mathbf{y}_u &= (P^{-1}_{\bar{p}} + H^T_{\bar{p}u}\bar{R}^{-1}_uH_{\bar{p}u})^{-1}H^T_{\bar{p}u}\bar{R}^{-1}_u\mathbf{y}_u \\ &= P_{\bar{p}}H^T_{\bar{p}u}(R_u + H_{\bar{p}u}P_{\bar{p}}H^T_{\bar{p}u})^{-1} \\ &= \bar{K}\mathbf{y}_u\end{aligned}$$

Thus the estimate of $\bar{\mathbf{x}}^{\bar{p}}$ is

$$\hat{\bar{\mathbf{x}}}^{\bar{p}}_{k+1|k+1} = \hat{\bar{\mathbf{x}}}^p_{k+1|k} + \bar{K}(\mathbf{y}_u - H_{\bar{p}u}\hat{\bar{\mathbf{x}}}^p_{k+1|k}) \tag{23}$$

Combining the results in (21) and (23) leads to the final form of the l_1-norm filter state estimates given in (15):

$$\hat{\mathbf{x}}_{k+1|k+1} = \begin{bmatrix} \hat{\bar{\mathbf{x}}}^{\bar{p}}_{k+1|k+1} \\ \hat{\bar{\mathbf{x}}}^p_{k+1|k+1} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{e}}^{p,nz} \end{bmatrix}$$

$$= \begin{bmatrix} \hat{\hat{\mathbf{x}}}^{p}_{k+1|k} + \bar{K}(\mathbf{y}_u - H_{\bar{p}u}\hat{\hat{\mathbf{x}}}^{p}_{k+1|k}) \\ \left(H_{pu}^T R_u^{-1} H_{pu}\right)^{-1} H_{pu}^T R_u^{-1}(\mathbf{y}_u - H_{\bar{p}u}\hat{\mathbf{x}}^{\bar{p}}_{k+1|k+1}) \end{bmatrix}$$

This proves the state estimate portion of the theorem.

The matrices $P^{e^p}_{k+1|k+1}$, $P_{\bar{x}e}$, and $P_{e\bar{x}}$ in (16) can be extracted from $P^{z}_{k+1|k+1}$ using the blockwise inversion formula. After substituting in the values of BD^{-1}, $D^{-1}C$ and D^{-1} the sum of $P^{e^p}_{k+1|k+1}$, $P_{\bar{x}e}$, and $P_{e\bar{x}}$ is

$$P^{\mathbf{e}^p}_{k+1|k+1} + P_{\bar{x}e} + P_{e\bar{x}} = \begin{bmatrix} \mathbf{0} & -P^{\bar{x}}_{\bar{p}p} - P^{\bar{x}}_{\bar{p}}\bar{H}^T_{\bar{p}u} \\ -P^{\bar{x}}_{p\bar{p}} - \bar{H}_{\bar{p}u}P^{\bar{x}}_{\bar{p}} & -P^{\bar{x}}_{p} + (H_{pu}^T R_u^{-1} H_{pu})^{-1} + \bar{H}_{\bar{p}u}P^{\bar{x}}_{\bar{p}}\bar{H}^T_{\bar{p}u} \end{bmatrix} \tag{24}$$

Substituting (24) into (16) gives the expression for $P_{k+1|k+1}$ in the theorem:

$$\begin{aligned} P_{k+1|k+1} &= \begin{bmatrix} P^{\bar{x}}_{\bar{p}} & P^{\bar{x}}_{\bar{p}p} \\ P^{\bar{x}}_{p\bar{p}} & P^{\bar{x}}_{p} \end{bmatrix} + P^{\mathbf{e}^p}_{k+1|k+1} + P_{\bar{x}e} + P_{e\bar{x}} \\ &= \begin{bmatrix} P^{\bar{x}}_{\bar{p}} & -P^{\bar{x}}_{\bar{p}}(H^{-L}_{pu}H_{\bar{p}u})^T \\ -H^{-L}_{pu}H_{\bar{p}u}P^{\bar{x}}_{\bar{p}} & (H_{pu}^T R_u^{-1} H_{pu})^{-1} + H^{-L}_{pu}H_{\bar{p}u}P^{\bar{x}}_{\bar{p}}(H^{-L}_{pu}H_{\bar{p}u})^T \end{bmatrix} \end{aligned}$$

□

There are several conclusions about the behavior and performance of the l_1-norm filter that can be drawn from Theorem 1. First, notice that the estimate of $\mathbf{x}_p$ in (19) is in fact a least-squares estimate given $\hat{\hat{\mathbf{x}}}^{\bar{p}}_{k+1|k+1}$ and the uncorrupted measurements, $\mathbf{y}_u$. In other words, the l_1-norm filter is re-initializing the estimate of $\mathbf{x}_p$ using the current set of uncorrupted measurements. Additionally, note that the estimates and covariance do not depend on the measurements corrupted by $\mathbf{e}^m_{k+1}$. This can be seen by observing that the updates do not include any terms that involve $\mathbf{y}_c$, H_c, and R_c. Thus, the same estimates can be reached by discarding the measurements that correspond to non-zero $\mathbf{e}^m_{k+1}$ detections before the joint state and error estimation step. These observations also indicate that the performance of the l_1-norm filter, for the case when only measurement outliers are present, should be comparable to a KF that discards measurements with residuals that exceed a χ^2 threshold. This behavior is verified using Monte Carlo simulations in Section IV.

In addition, when all sparse errors are correctly detected, the l_1-norm filter estimates are unbiased. The proof of this result will require the following lemma.

Lemma 2. $\bar{K}H_{pu} = \mathbf{0}$

Proof. First note that

$$\begin{aligned}\bar{R}_u^{-1}H_{pu} &= R_u^{-1}H_{pu} - R_u^{-1}H_{pu}(H_{pu}^T R_u^{-1}H_{pu})^{-1}H_{pu}^T R_u^{-1}H_{pu} \\ &= R_u^{-1}H_{pu} - R_u^{-1}H_{pu} = \mathbf{0}\end{aligned}$$

Applying a Schur identity [29] shows that

$$\left(\bar{R}_u + H_{\bar{p}u}P_{\bar{p}}H_{\bar{p}u}^T\right)^{-1} = \bar{R}_u^{-1} - \bar{R}_u^{-1}H_{\bar{p}u}(P_{\bar{p}}^{-1} + H_{\bar{p}u}R_u^{-1}H_{\bar{p}u}^T)^{-1}H_{\bar{p}u}^T\bar{R}_u^{-1}$$

Combining these results shows that

$$\begin{aligned}\bar{K}H_{pu} &= P_{\bar{p}}H_{\bar{p}u}^T\left(\bar{R}_u + H_{\bar{p}u}P_{\bar{p}}H_{\bar{p}u}^T\right)^{-1}H_{pu} \\ &= P_{\bar{p}}H_{\bar{p}u}^T(\bar{R}_u^{-1}H_{pu} - \bar{R}_u^{-1}H_{\bar{p}u}(P_{\bar{p}}^{-1} + H_{\bar{p}u}R_u^{-1}H_{\bar{p}u}^T)^{-1}H_{\bar{p}u}^T\bar{R}_u^{-1}H_{pu}) = \mathbf{0}\end{aligned}$$

□

Theorem 2. *If Step 2 of Algorithm 1 detects all non-zero components of* $\mathbf{e}_k^p$ *and* $\mathbf{e}_{k+1}^m$ *then* $\hat{\mathbf{x}}_{k+1|k+1}$ *is unbiased.*

Proof. Using (19), the posterior state estimation error is

$$\begin{aligned}\tilde{\mathbf{x}}_{k+1|k+1} &= \begin{bmatrix}\tilde{\mathbf{x}}_{k+1|k+1}^{\bar{p}} \\ \tilde{\mathbf{x}}_{k+1|k+1}^{p}\end{bmatrix} \\ &= \begin{bmatrix}\mathbf{x}_{k+1}^{\bar{p}} - \hat{\mathbf{x}}_{k+1|k}^{\bar{p}} + \bar{K}(\mathbf{y}_u - H_{\bar{p}u}\hat{\mathbf{x}}_{k+1|k}^{\bar{p}}) \\ \mathbf{x}_{k+1}^{p} - \left(H_{pu}^T R_u^{-1}H_{pu}\right)^{-1}H_{pu}^T R_u^{-1}(\mathbf{y}_u - H_{\bar{p}u}\hat{\mathbf{x}}_{k+1|k+1}^{\bar{p}})\end{bmatrix} \\ &= \begin{bmatrix}(I - \bar{K}H_{\bar{p}u})\tilde{\mathbf{x}}_{k+1|k}^{\bar{p}} + \bar{K}(H_{pu}\mathbf{x}_{k+1}^{p} + \mathbf{v}_{k+1}^{u}) \\ -\left(H_{pu}^T R_u^{-1}H_{pu}\right)^{-1}H_{pu}^T R_u^{-1}(H_{pu}\tilde{\mathbf{x}}_{k+1|k+1}^{\bar{p}} + \mathbf{v}_{k+1}^{u})\end{bmatrix}\end{aligned} \tag{25}$$

After applying Lemma 2, $\tilde{\mathbf{x}}_{k+1|k+1}^{\bar{p}}$ can be shown to be

$$\tilde{\mathbf{x}}_{k+1|k+1}^{\bar{p}} = (I - \bar{K}H_{\bar{p}u})\tilde{\mathbf{x}}_{k+1|k}^{\bar{p}} + \bar{K}\mathbf{v}_{k+1}^{u}$$

and thus $E\left[\tilde{\mathbf{x}}_{k+1|k+1}^{\bar{p}}\right] = \mathbf{0}$. Moreover this implies that $E\left[\tilde{\mathbf{x}}_{k+1|k+1}^{p}\right] = \mathbf{0}$. □

In the case where the error detection worked perfectly, these results indicate that the l_1-norm filter performs as desired: it ignores faulty measurements that could negatively impact the state estimates and it corrects erroneous state estimates. But since the l_1-norm solution is not guaranteed to correctly detect $\mathbf{e}^m_{k+1}$ and $\mathbf{e}^p_k$ it is also important to evaluate the impact of incorrect error detection, either false alarms or missed detections.

In the case of false alarms (i.e. incorrectly detecting an error when it is not present), the l_1-norm estimates will still be unbiased, the only cost will be an increased posterior state covariance.

Theorem 3 (False Alarm Case)**.** *If there are false alarms in Step 2 of Algorithm 1 then* $\hat{\mathbf{x}}_{k+1|k+1}$ *will be unbiased. However,* $P_{k+1|k+1} \geq P^{opt}_{k+1|k+1}$*, where* $P^{opt}_{k+1|k+1}$ *is the posterior covariance if no false alarms had occurred.*

Proof. If there are false alarms, the residuals will take the same form as (25) and thus the estimates will remain unbiased.

To simplify the covariance portion of the proof, $\mathbf{e}^m_{k+1}$ and $\mathbf{e}^p_k$ false alarms will be handled separately. If there are $\mathbf{e}^m_{k+1}$ false alarms the information matrix without false alarms and the l_1-norm filter information matrix will take the following forms:

$$\Lambda^{opt}_{k+1|k+1} = \Lambda_{k+1|k} + H_u^T R_u^{-1} H_u + H_c^T R_c^{-1} H_c$$

$$\Lambda^{l_1}_{k+1|k+1} = \Lambda_{k+1|k} + H_u^T R_u^{-1} H_u$$

Taking the difference of the two shows that

$$\Lambda^{opt}_{k+1|k+1} - \Lambda^{l_1}_{k+1|k+1} = H_c^T R_c^{-1} H_c > 0$$

Which implies that $\Lambda^{opt}_{k+1|k+1} > \Lambda^{l_1}_{k+1|k+1}$ which in turn implies that $P^{opt}_{k+1|k+1} < P^{l_1}_{k+1|k+1}$.

If there are $\mathbf{e}^p_k$ false alarms the information matrix without false alarms and the l_1-norm filter information matrix will take the following forms:

$$\Lambda^{opt}_{k+1|k+1} = \begin{bmatrix} \Lambda_{\bar{p}} + H_{\bar{p}}^T R^{-1} H_{\bar{p}} & \Lambda_{\bar{p}p} + H_{\bar{p}}^T R^{-1} H_p \\ \Lambda_{p\bar{p}} + H_p^T R^{-1} H_{\bar{p}} & \Lambda_p + H_p^T R^{-1} H_p \end{bmatrix}$$

$$\Lambda^{l_1}_{k+1|k+1} = \begin{bmatrix} \Lambda_{\bar{p}} + H_{\bar{p}}^T R^{-1} H_{\bar{p}} & H_{\bar{p}}^T R^{-1} H_p \\ H_p^T R^{-1} H_{\bar{p}} & H_p^T R^{-1} H_p \end{bmatrix}$$

Taking the difference of the two shows that

$$\Lambda^{opt}_{k+1|k+1} - \Lambda^{l_1}_{k+1|k+1} = \begin{bmatrix} \mathbf{0} & \Lambda_{\bar{p}p} \\ \Lambda_{p\bar{p}} & \Lambda_p \end{bmatrix} \geq 0$$

Which implies that $\Lambda^{opt}_{k+1|k+1} \geq \Lambda^{l_1}_{k+1|k+1}$ which in turn implies that $P^{opt}_{k+1|k+1} \leq P^{l_1}_{k+1|k+1}$. □

Another interpretation of this theorem is that false alarms in the error detection step will reduce the amount of information available to the filter to reduce the covariance. If the false detections are measurement errors, the filter will ignore those measurements and thus will lose the ability to reduce the state covariance with those measurements. If the false detections are state prediction errors, the prior information about those states will be ignored and as a result the covariance for those states will be larger.

Finally, if there are missed error detections the following theorem demonstrates that the l_1-norm filter estimates will be biased.

Theorem 4 (Missed Detection Case)**.** *If any non-zero terms in $\mathbf{e}^p_k$ or $\mathbf{e}^m_{k+1}$ are not detected in Step 2 of Algorithm 1, then $\hat{\mathbf{x}}^{k+1|k+1}$ will be biased. In the worst case where no outliers are detected, the bias will be equal to $\mathbf{b}^{KF}$, the bias of the Kalman filter estimates.*

Proof. Let $\mathbf{e}^p_{\bar{p}}$ and $\mathbf{e}^m_u$ be the undetected state prediction and measurement outliers respectively. Then the posterior state estimation error will be

$$\tilde{\mathbf{x}}_{k+1|k+1} = \begin{bmatrix} (I - \bar{K}H_{\bar{p}u})(\tilde{\mathbf{x}}^{\bar{p}}_{k+1|k} + \mathbf{e}^p_{\bar{p}}) + \bar{K}(\mathbf{v}^u_{k+1} + \mathbf{e}^m_u) \\ -\left(H^T_{pu}R^{-1}_u H_{pu}\right)^{-1} H^T_{pu}R^{-1}_u(H_{pu}\tilde{\mathbf{x}}^{\bar{p}}_{k+1|k+1} + \mathbf{v}^u_{k+1} + \mathbf{e}^m_u) \end{bmatrix} \tag{26}$$

Taking the expected value of $\tilde{\mathbf{x}}_{k+1|k+1}$ shows that

$$E\left[\tilde{\mathbf{x}}_{k+1|k+1}\right] = \begin{bmatrix} (I - \bar{K}H_{\bar{p}u})\mathbf{e}^p_{\bar{p}} + \bar{K}\mathbf{e}^m_u \\ \left(H^T_{pu}R^{-1}_u H_{pu}\right)^{-1} H^T_{pu}R^{-1}_u(H_{\bar{p}u})(I - \bar{K}H_{\bar{p}u})\mathbf{e}^p_{\bar{p}} + (I + H_{\bar{p}u}\bar{K})\mathbf{e}^m_u \end{bmatrix} \tag{27}$$

thus the estimates are biased.

If no outliers are detected, then

$$\tilde{\mathbf{x}}_{k+1|k+1} = \tilde{\mathbf{x}}^{\bar{p}}, \quad H_{\bar{p}u} = H_{k+1} \quad \text{and} \quad \bar{K} = P_{k+1|k}H_{k+1}\left(H_{k+1}P_{k+1|k}H^T_{k+1} + R_{k+1}\right)^{-1} = K_{KF}$$

where K_{KF} is the Kalman gain. Thus the bias is

$$b_{l_1} = (I - K_{KF}H_{k+1})\mathbf{e}^p_k + K_{KF}\mathbf{e}^m_{k+1} \tag{28}$$

The Kalman filter residuals in this case would be

$$\tilde{\mathbf{x}}_{KF} = (I - K_{KF}H_{k+1})(\tilde{\mathbf{x}}_{k+1|k} + \mathbf{e}_k^p) + K_{KF}(\mathbf{v}_{k+1} + \mathbf{e}_{k+1}^m) \tag{29}$$

Taking the expected value of (29) shows that the Kalman filter bias is

$$\mathbf{b}_{KF} = (I - K_{KF}H_{k+1})\mathbf{e}_k^p + K_{KF}\mathbf{e}_{k+1}^m = b_{l_1} \tag{30}$$

□

While returning biased state estimates is an issue, note that the bias can be absorbed into the $\mathbf{e}_k^p$ term and detected and corrected by the l_1-norm filter at the next measurement update. Also note that if none of the outliers were detected (the worst-case scenario) the bias in $\hat{\mathbf{x}}_{k+1|k+1}$ will be no worse than the bias for the Kalman filter estimates. These results also indicate that it is preferable to set the χ^2 threshold τ conservatively (i.e. choosing a value that corresponds to a 95% confidence interval rather than a 99% confidence interval) because it will reduce the likelihood of biasing the state estimates. Finally, unless the measurement noise and sparse error terms are pathologically adversarial (e.g. a set of large outliers are exactly canceled by the measurement and process noise and thus rendered undetectable), a missed error detection should typically correspond to a small error that would be difficult for any detection scheme to distinguish from the process and measurement noise.

F. Application to Nonlinear Systems

The l_1-norm filter can also be applied to nonlinear systems using an extended Kalman filter (EKF) based algorithm. In that case, the matrices H_{k+1} and F_k will be the Jacobians of the nonlinear measurement and state propagation functions, respectively, evaluated at the current state estimate. Additionally, the information filter update in step 5 of Algorithm 1 should be replaced with an extended information filter update.

It should also be noted that, as with the EKF, the theoretical guarantees for the nonlinear version of the l_1-norm filter are not as strong. For instance, it can not be guaranteed that the state estimates will be unbiased because of the impact of linearization errors. However, the Monte Carlo simulations and experimental results in the next two sections demonstrate that the nonlinear

version of the l_1-norm filter can provide superior state estimation performance relative to other state-of-the-art robust filtering algorithms.

IV. Monte Carlo Simulation Results

A simulated target tracking scenario is used in this section to evaluate the performance of the l_1-norm filter in the presence of unmodeled measurement and state prediction errors. Monte Carlo trials were run for three different cases: scenarios with measurement errors only, scenarios with state prediction errors and finally scenarios with simultaneous measurement and state prediction errors. Several other filtering algorithms were also evaluated in order to demonstrate the improved performance of the l_1-norm filter, especially in cases where both unmodeled state prediction and measurement errors occur.

A. Simulation Setup

The Monte Carlo trials simulate a 2D single target tracking scenario with position and velocity measurements. The estimated states were the target position and velocity. Four independent sets of position and velocity measurements of the target were simulated with an update rate of 1 Hz. The target dynamics were simulated using a constant velocity model [2] and the total length of each Monte Carlo trial was 30 seconds. The nominal process and measurement noise covariances were

$$Q_k = 0.1 \begin{bmatrix} \Delta t^4/4 & 0 & \Delta t^3/2 & 0 \\ 0 & \Delta t^4/4 & 0 & \Delta t^3/2 \\ \Delta t^3/2 & 0 & \Delta t & 0 \\ 0 & \Delta t^3/2 & 0 & \Delta t \end{bmatrix}, \quad \sigma^2_{range} = 1m, \quad \sigma^2_{range-rate} = 0.1m/s$$

where $\Delta t = 1s$ is the propagation time between sets of measurements.

Measurement errors were simulated by sampling the measurement noise for the position (velocity) measurements from a Gaussian distribution with a mean of 30 meters (1 meters/sec.) instead of the nominal zero mean distribution. State prediction errors were induced by sampling the process noise from a Gaussian distribution with a larger covariance ($\bar{Q} = 10 \cdot Q_k$) than the nominal process noise model while generating the target trajectory. For the simulations with both state prediction and measurement outliers, the number of measurement outliers at any time step was chosen to

ensure that the error sparsity requirements for the l_1-norm filter were met (i.e. the dimension of the non-zero state prediction and measurement errors were less than or equal to the number of measurements).

For each set of Monte Carlo trials, the performance of the l_1-norm filter was compared against the Kalman Filter (KF), unscented Kalman filter (UKF) [30], a robust statistics based Kalman filter (RKF) [4, 6], and a variational Bayes robust filter (VBAKF) [7]. The χ^2 threshold parameter τ for the l_1-norm filter was set to 15.5073, which corresponds to a 95% confidence interval for the χ^2 test. The robust cost function for the RKF was chosen so that it was equivalent to a KF that discards measurements with residuals that fail a χ^2-test. The threshold for the RKF χ^2-test was set to match the χ^2 thresholds used in the l_1-norm filter so that if only measurement errors are present the RKF and l_1-norm filter will identify the same set of corrupted measurements.

B. Measurement Error Only Results

The first set of simulations focused on assessing the performance of the l_1-norm filter when errors were present in the measurements only. The percentage of measurements that were corrupted with the off-nominal noise was varied from 0 to 100% in increments of 10%. For each percentage level, 100 Monte Carlo trials were performed with the corrupted measurements chosen uniformly at random.

The average position error as a function of the percentage of corrupted measurements is shown in Figure 1. Error bars were left off of the UKF and EKF results to preserve the clarity of the plot. As the number of measurement outliers increases, the performance of the non-robust filters (EKF and UKF) degrades significantly. In contrast, the robust approaches are able to maintain reasonable average positioning errors even as all of the measurements are corrupted with errors. Additionally, these plots empirically verify that the l_1-norm filter and the RKF performance are similar when only measurement outliers are present.

Finally, the average probability of detection (P_d) and probability of false alarm (P_{fa}) for $\mathbf{e}_{k+1}^m$ were 0.9996 and 0.0 respectively. The average P_{fa} for $\mathbf{e}_k^p$ was 0.008 and the majority of the false alarms can be attributed to correcting biases introduced by missed $\mathbf{e}_{k+1}^m$ detections.

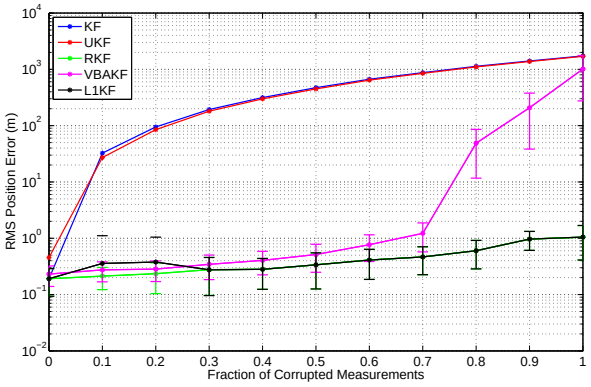

Fig. 1 Average position error vs. fraction of corrupted measurements

C. Process Error Only Results

The next set of simulations focused on assessing the performance of the l_1-norm filter when errors were present in the state predictions only. The percentage of state updates that were corrupted with the off nominal noise was varied from 0 to 100% in increments of 10%. For each percentage level, 100 Monte Carlo trials were performed with the corrupted state updates chosen uniformly at random.

The average position error as a function of the percentage of process errors is shown in Figure 2. Error bars were left off of the RKF and VBAKF results to preserve the clarity of the plot. The l_1-norm filter results and EKF results are nearly identical in this case and correspond to the line at the bottom of the plot. In contrast to the measurement error only results, the EKF and UKF outperform all of the robust filters (with the exception of the l_1-norm filter) even when only a small fraction of the state updates are corrupted with additional noise. In this case, the error models for the RKF and VBAKF are not adequate to compensate for the additional noise because neither algorithm accounts for additional errors in the process model beyond the nominal process noise. The l_1-norm filter explicitly models for both process and measurement errors and thus is able to correct for the additional process noise when it is present.

For this example, P_d and P_{fa} for $\mathbf{e}_k^p$ were 0.12 and 0.0 respectively. There were no $\mathbf{e}_{k+1}^m$ false alarms. The low P_d values can in part be attributed to the distribution chosen for $\mathbf{e}_k^p$, which was zero-mean but had a larger covariance than the nominal process noise. At least some of the samples drawn from that distribution would be consistent with the nominal process noise and thus difficult to detect. These results indicate that correcting for the largest state prediction errors (i.e. the ones

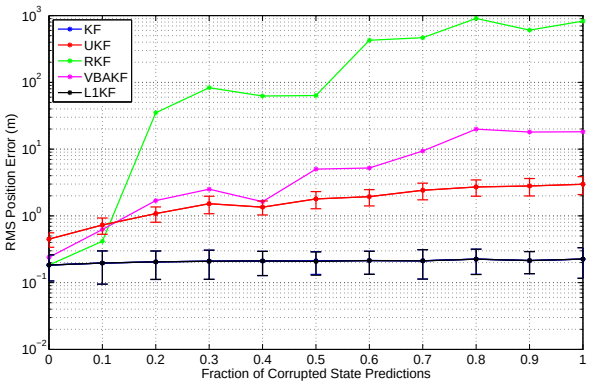

Fig. 2 Average position error vs. fraction of process errors

most likely to be detected) provides a significant performance gain. In addition, they bolster the claim that when there are missed detections in the l_1-norm filter they often correspond to errors that are small relative to the measurement and process noise and thus will have limited impact on the state estimates.

D. Combined Measurement and Process Error Results

The final set of simulations focused on assessing the performance of the l_1-norm filter when errors were present in both the state predictions and measurements. In this case, the percentage of state updates that were subject to the off nominal noise and the percentage of measurement errors were varied together (i.e. 10% of measurements were corrupted and 10% of state updates were corrupted for the same set of Monte Carlo trials). For each percentage level, 100 Monte Carlo trials were performed with the corrupted measurement and state updates chosen uniformly at random. The simulations were only run up to 80% error corruption because after that the error sparsity assumption could not be satisfied.

The average position error as a function of the percentage of process errors is shown in Figure 3. This set of trials represents a worst case scenario that only the l_1-norm filter can handle. The EKF and UKF estimates are not robust to the measurement errors and thus have large state estimation errors while the RKF and VBAKF can not correctly compensate for the process errors. Only the l_1-norm filter is able to correctly compensate for both the state prediction and measurement errors when they occur simultaneously and is able to maintain reasonable performance even when the

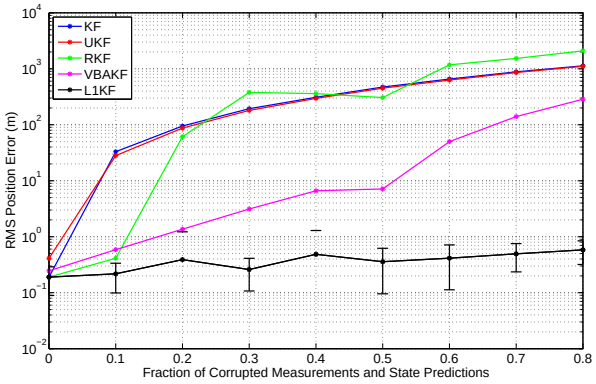

Fig. 3 Average position error vs. fraction of process and measurement errors

majority of the state predictions and measurements are incorrect.

For this example, P_d and P_{fa} for $\mathbf{e}_{k+1}^m$ were 0.9996 and 0.0 respectively. P_d and P_{fa} for $\mathbf{e}_k^p$ were 0.15 and 0.005 respectively. As with the measurement error only case, the majority of the $\mathbf{e}_k^p$ false alarms can be attributed to correcting biases introduced by missed $\mathbf{e}_{k+1}^m$ detections.

Overall these Monte Carlo simulations show that the l_1-norm filter can provide robust state estimates over a broader range of conditions than other robust filtering algorithms. In situations where only measurement errors are present, the l_1-norm filter can match the performance of state-of-the-art robust filtering algorithms. For situations with state prediction outliers the l_1-norm filter can provide superior performance to other robust filtering approaches because it explicitly models state predictions errors while the other algorithms do not.

V. Vision-Aided Navigation Experimental Results

This section presents experimental results demonstrating the performance of the l_1-norm filter applied to vision-aided navigation in an urban area. In the data collected, GPS measurements were corrupted intermittently with multipath, while the state predictions were corrupted by drift from visual odometry measurements. Three other filtering approaches (the EKF, and two robust filtering techniques) are compared to the performance of the l_1-norm filter. This experiment demonstrates that the l_1-norm filter is able to outperform the other algorithms because it can compensate for both the GPS measurement errors and the accumulated state prediction errors from the visual odometry measurements.

A. Vision-aided Navigation Background

Vision-aided navigation focuses on how to fuse visual information captured from a camera with other sensors to localize a vehicle in a global coordinate system.

In recent years, vision-aided navigation has been demonstrated on a number of platforms. Often these systems fuse visual odometry with other sensors (IMU, GPS, LiDAR) to generate a global state estimate. Visual odometry has several error sources that can impact the accuracy of these state estimates. Most notably, a bias is introduced by long range features in stereo visual odometry [31, 32]. Scale factor and misalignment errors in the visual odometry data can also occur and cause the navigation solution to drift over time [33].

Although recursive filtering approaches to vision-aided navigation have been developed [34], many current approaches use optimization-based pose graph estimation techniques to generate the navigation solutions [31, 35]. Recent research has shown that optimization based approaches to vision-aided navigation can outperform recursive filtering algorithms for a number of applications [36]. One reason that pose graph optimization tends to perform better than filtering is that previous poses can be updated each time the optimization is solved, thus allowing errors in previous pose estimates to be corrected, leading to a more accurate positioning solution at the current time. In contrast, filtering algorithms can not retroactively change previous state estimates in an efficient way because the estimates are marginalized out at each measurement update. Thus, any state estimation errors made earlier in the filter will propagate forward to future state estimates.

The l_1-norm filter tackles this problem by detecting situations when the current state estimate is inconsistent with the current set of measurements. After detecting these situations, the filter adjusts the state estimates to account for the impact of state estimation error that has been propagated to the current time step. In this way, the l_1-norm filter can adjust its state estimates when drift errors accumulate without having to resolve for any of its previous estimates.

B. Vision-aided Navigation with the l_1-norm Filter

There were several challenges associated with using both vision and GPS measurements in the l_1-norm filter. First, the GPS and visual odometry data were not being generated at the same

rate (10 Hz for the vision vs. 1 Hz for the GPS). In practice, large errors in the visual odometry are not observable unless there is additional information from another measurement such as GPS. Thus, the majority of the visual odometry measurements could not be checked for errors directly by the l_1-norm filter. Additionally, it was found that the errors in the visual odometry data were often below the detection threshold of the l_1-norm filter for any given measurement even when GPS measurements were available. Fortunately, it was determined that the cumulative effects of the visual odometry errors (over several sets of measurements) were large enough and could be detected by the l_1-norm filter as state propagation errors, $\mathbf{e}_k^p$, when GPS measurements were available.

One more step had to be added to the l_1-norm filter procedure to provide reasonable navigation performance with the visual odometry data. The position states were not directly observable using the visual odometry data and as a result their covariance grew substantially in between GPS measurement updates. Since the state covariance was used in the thresholding step (Step 2 in Algorithm 1) in the l_1-norm filter to determine which errors were non-zero, this often led to situations where large values of $\hat{\mathbf{e}}_k^p$ were being thresholded out and ignored, which then caused large state estimation errors. To repair this problem, an additional criterion was added to the thresholding step. If a component of $\hat{\mathbf{e}}_k^p$ was above a threshold level, then that component was estimated using the information filter regardless of the outcome of the χ^2 thresholding step.

C. Experimental Setup

The data used for this experiment was collected while driving along roads in the Boston area. Environments driven through varied between dense urban canyons and areas of good GPS coverage along the Charles River. The total time for the experiment took approximately 25 minutes from start to finish and the total distance covered was 7.32 km. Vehicle speeds varied between 0 and 72 km/h. The estimated states were the car's position and velocity in Earth-Centered Earth-Fixed coordinates.

The sensors used for the experiment were a dashboard-mounted stereo vision camera (Point Grey BumbleBee2 with a resolution of 512 x 384 and 43° field of view) and a consumer grade GPS receiver (uBlox EVK-6T). Visual odometry measurements (measuring the change in position of the

car between camera frames) and GPS pseudoranges were processed in the navigation filter. Visual odometry measurements were provided at 10 Hz while the GPS receiver reported pseudoranges at 1 Hz when they were available. More details about the system used for the experimental data can be found in [37]. A high accuracy GPS positioning solution that was generated by the receiver was used as ground-truth for the experiment.

The pseudoranges and state predictions were compared against the truth data to verify that the error sparsity assumptions were satisfied. These comparisons indicate that, during the experiment, at most 2 pseudorange measurements were corrupted with multipath at each time step and that, when multipath errors occurred, there were at least 6 total pseudorange measurements available. When large state prediction errors occurred (i.e., $\mathbf{e}^p \neq \mathbf{0}$), at least 6 pseudorange measurements were available. Additionally, the results showed that simultaneous state prediction and multipath errors never occurred. Therefore, during the experiment, the error sparsity requirements of the l_1-norm filter were satisfied because the number of available measurements was always larger than the number of non-zero entries of the sparse errors.

D. Experimental Results

The experimental data was processed using the l_1-norm filter as well as three other algorithms to compare the performance of each in an urban navigation scenario. The visual odometry measurements were modeled using the stochastic cloning technique proposed by Roumeliotis et al. [38]. The vehicle dynamics were modeled using a constant velocity model [2].

The three other algorithms were an EKF, a VB robust filter called the outlier robust Kalman filter (ORKF) [9, 15], and an EKF that uses robust statistics to reduce the impact of measurement outliers. In the experimental results, the last filter will be referred to as the robust Kalman filter (RKF) and is similar to algorithms presented by Masreliez and Martin [4] and Schick and Mitter [6]. Since the EKF is not a robust estimator and the experimental dataset contains both GPS multipath errors and visual odometry drift and bias errors, the robust filtering algorithms should produce better results than the EKF.

A comparison of the positioning error of the navigation solutions for the four algorithms is shown

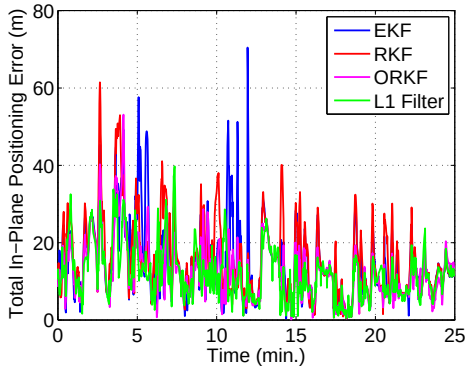

Fig. 4 Positioning error vs. time for each of the algorithms

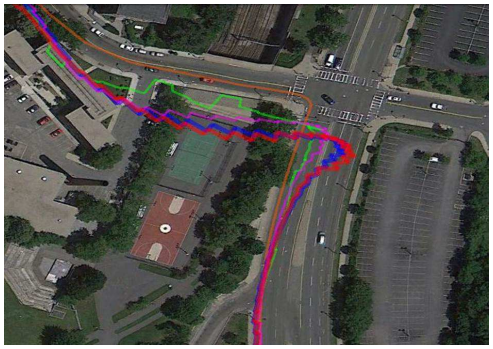

Fig. 5 Impact of visual odometry errors on the RKF (shown in red) and ORKF (shown in magenta) solutions. In this case, the turning motion of the car at the intersection induced errors in the visual odometry solution because most of the features that were being tracked left the field of view of the camera. GPS truth in this figure is shown in orange.

in Figure 4. The RKF solution has a number of instances where the positioning error exceeds all of the other algorithms by a significant amount. These large errors are primarily caused by the fact that the RKF can not distinguish between *a priori* state estimation errors (in this case caused by errors accumulated from the visual odometry measurements) and GPS measurement errors. For instance, the large deviation from truth shown in Figure 5 is the result of accumulated visual odometry errors that occurred, when the vehicle turned at the intersection. In this case, the turning motion of the car induced errors in the visual odometry solution because most of the features that were being tracked left the field of view of the camera. The GPS measurement residuals became large and as a result, the RKF significantly downweighted GPS measurements that could have been used to correct for

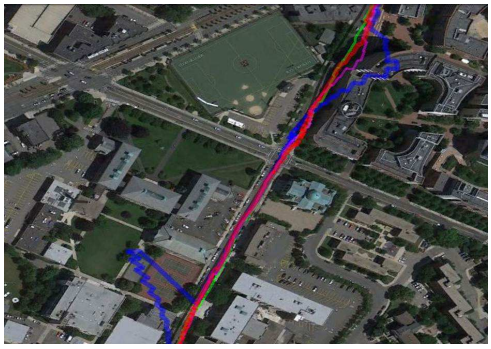

Fig. 6 Impact of multipath on the EKF navigation solution (shown in blue). The multipath errors are caused by reflections off of the tall buildings near the road. The ORKF (shown in magenta), RKF (shown in red) and l_1-norm filter (shown in green) were able to detect and compensate for the impact of the multipath in this case.

the accumulated visual odometry errors in the state estimates. In the case shown in Figure 5, the ORKF also takes more time than the EKF and l_1-norm filter to recover from the visual odometry errors because it can not differentiate between the state propagation errors and GPS measurement errors and also ends up downweighting GPS measurements that could help the filter converge to the correct solution.

In contrast, the l_1-norm filter was able to determine that the large measurement residuals were the results of *a priori* state errors instead of GPS measurement errors and as a result, was able to use the GPS measurements to recover from the visual odometry errors. The EKF was not significantly affected in these situations because even though visual odometry errors have accumulated in the state estimates, processing the GPS measurements quickly corrects for the impact of the error because the measurement residuals are so large.

Upon examining the EKF results, there are several large positioning errors around 11 minutes into the experiment. These are the result of multipath errors in the GPS pseudorange measurements caused by a large building (see Figure 6 for a more detailed view). In this case, all of the robust filters were able to detect and eliminate the impact of the multipath on the navigation solution as expected.

Summary statistics for all of the algorithms are shown in Table 1. Based on this dataset,

Table 1 Comparison of Positioning Error Results

Algorithm	Mean Error (m)	σ Error (m)	Max Error (m)	Error Relative to EKF (m)
EKF	15.10	10.16	70.45	0.0
RKF [4, 6]	15.53	9.66	61.47	0.43
ORKF [9, 15]	12.38	7.51	53.10	-2.72
l_1-norm Filter	**12.00**	**6.87**	**39.76**	**-3.10**

the l_1-norm filter is able to provide the best solution out of the four algorithms. It is able to provide accurate state estimates when the GPS measurements are corrupted with multipath and avoids incorrectly ignoring GPS as the ORKF and RKF do when significant visual odometry errors accumulate. Additionally, the l_1-norm filter has the ability to perform state estimation reliably when both of these situations occur simultaneously, which none of the other algorithms can guarantee.

The absolute position errors shown in Table 1 are larger than one might expect from a navigation solution based in part on GPS data. In this experiment, additional corrections for errors in the pseudoranges due to ionospheric effects (i.e. corrections generated by the Wide-Area Augmentation System (WAAS)) were unavailable and as a result the pseudoranges were biased. Although errors due to ionspheric delays could have been corrected using WAAS data, localized errors in the pseudoranges such as GPS multipath could not have been compensated for and would still have been present. Additionally, the WAAS corrections would not have had an impact on the visual odometry errors that occurred. Thus, while using WAAS corrections would have reduced the absolute error for all of the algorithms, the reductions in error relative to the EKF (the final column of Table 1) would still have occurred because they are related to the compensation of multipath and visual odometry errors that the WAAS corrections could not fix.

VI. Conclusions and Future Work

This paper presented a recursive state estimation algorithm, the l_1-norm filter, that improves robustness to both unmodeled state prediction and measurement errors. The l_1-norm filter detects the presence of unmodeled errors using a convex optimization. Given that information, the filter can then adjust the *a priori* state estimates and measurements accordingly to compensate for the

errors. The algorithm is also computationally efficient as it combines a convex optimization with standard recursive filtering steps.

A simulated target tracking scenario was used to evaluate the performance of the l_1-norm filter and compare it to existing state of the art robust state estimation algorithms. The l_1-norm filter was also evaluated on a dataset consisting of visual odometry and GPS data collected in urban areas around Boston. In both cases, the l_1-norm filter was able to outperform state-of-the-art robust state estimation algorithms, because it could compensate for both state prediction and measurement outliers that occurred in the data.

It would be worthwhile to investigate whether the hard thresholds that are set in the l_1-norm filter for detecting unmodeled state estimation and measurement errors can be adapted online so that a poor initialization of the thresholds does not impact the filter performance. Finally, testing the algorithm on datasets with more sensors to determine the broader applicability of the sparse error model used to develop the l_1-norm filter would be beneficial.

Acknowledgments

This work was funded by the C. S. Draper Laboratory Internal Research and Development Program.

References

[1] Gelb, A., editor, *Applied Optimal Estimation*, MIT Press, 1974.

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

[3] Kaplan, E. D. and Hegarty, C. J., editors, *Understanding GPS: Principles and Applications*, Artech House, 2006.

[4] Masreliez, C. and Martin, R., "Robust Bayesian Estimation for the Linear Model and Robustifying the Kalman Filter," *IEEE Transactions on Automatic Control*, Vol. 22, No. 3, Jun 1977, pp. 361–371.

[5] Willsky, A. S., "A Survey of Design Methods for Failure Detection in Dynamic Systems," *Automatica*, Vol. 12, 1976, pp. 601–611.

[6] Schick, I. C. and Mitter, S. K., "Robust Recursive Estimation in the Presence of Heavy-Tailed Observation Noise," *Annals of Statistics*, Vol. 22, 1994, pp. 1045–1080.

[7] Särkkä, S. and Nummenmaa, A., "Recursive Noise Adaptive Kalman Filtering by Variational Bayesian Approximations," *IEEE Transactions on Automatic Control*, Vol. 54, No. 3, March 2009, pp. 596–600.

[8] Ting, J.-A., D'Souza, A., and Schaal, S., "Automatic Outlier Detection: A Bayesian Approach," *IEEE International Conference on Robotics and Automation*, April 2007, pp. 2489–2494.

[9] Agamennoni, G., Nieto, J. I., and Nebot, E. M., "Approximate Inference in State-Space Models with Heavy-Tailed Noise," *IEEE Transactions on Signal Processing*, Vol. 60, No. 10, Oct 2012, pp. 5024–5037.

[10] Graham, M., Steiner, T., and How, J., "Robust Vision-Aided Navigation in Urban Environments," *AIAA Guidance Navigation and Control Conference 2013*, August 2013.

[11] Tapley, B. D., Schutz, B. E., and Born, G. H., *Statistical Orbit Determination*, Elsevier Academic Press, 2004.

[12] Grimble, M. and El Sayed, A., "Solution of the H_∞ Optimal Linear Filtering Problem for Discrete-time Systems," *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 38, No. 7, Jul 1990, pp. 1092–1104.

[13] Shen, W. and Deng, L., "Game Theory Approach to Discrete H_∞ Filter Design," *IEEE Transactions on Signal Processing*, Vol. 45, No. 4, Apr 1997, pp. 1092–1095.

[14] Simon, D., *Optimal State Estimation: Kalman, H_∞, and Nonlinear Approaches*, Wiley, 2006.

[15] Agamennoni, G., Nieto, J. I., and Nebot, E. M., "An Outlier-Robust Kalman Filter," *IEEE International Conference on Robotics and Automation*, May 2011, pp. 1551–1558.

[16] Mattingley, J. and Boyd, S., "Real-Time Convex Optimization in Signal Processing," *IEEE Signal Processing Magazine*, Vol. 27, No. 3, May 2010, pp. 50–61.

[17] Kim, D., Lee, S.-G., and Jeon, M., "Outlier Rejection Methods for Robust Kalman Filtering," *Future Information Technology*, edited by J. J. Park, L. T. Yang, and C. Lee, Vol. 184 of *Communications in Computer and Information Science*, Springer Berlin Heidelberg, 2011, pp. 316–322.

[18] Candes, E. J. and Randall, P. A., "Highly Robust Error Correction by Convex Programming," *IEEE Transactions on Information Theory*, Vol. 54, No. 7, July 2008, pp. 2829–2840.

[19] Vaswani, N., "Kalman Filtered Compressed Sensing," *15th IEEE International Conference on Image Processing*, Oct 2008, pp. 893–896.

[20] Carmi, A., Gurfil, P., and Kanevsky, D., "Methods for Sparse Signal Recovery Using Kalman Filtering With Embedded Pseudo-Measurement Norms and Quasi-Norms," *IEEE Transactions on Signal Processing*, Vol. 58, No. 4, April 2010, pp. 2405–2409.

[21] Charles, A. S. and Rozell, C. J., "Dynamic filtering of sparse signals using reweighted l_1," *2013 IEEE*

International Conference on Acoustics, Speech and Signal Processing, May 2013, pp. 6451–6455.

[22] Mohiuddin, S., Gustafson, D. E., and Rachlin, Y., "Mitigating the Effects of GNSS Multipath with a Coded Filter," *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation*, 2011, pp. 2381–2394.

[23] Eldar, Y. and Kutyniok, G., editors, *Compressed Sensing: Theory and Applications*, Cambridge University Press, 2012.

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

[25] Bertsekas, D., *Convex Optimization Theory*, Athena Scientific, 2009.

[26] Candes, E. J., Wakin, M. B., and Boyd, S. P., "Enhancing Sparsity by Reweighted l_1 Minimization," *Journal of Fourier Analysis and Applications*, Vol. 14, No. 5, 2008, pp. 877–905.

[27] Figueiredo, M., Nowak, R., and Wright, S., "Gradient Projection for Sparse Reconstruction: Application to Compressed Sensing and Other Inverse Problems," *IEEE Journal of Selected Topics in Signal Processing*, Vol. 1, No. 4, Dec 2007, pp. 586–597.

[28] Maybeck, P. S., *Stochastic Models Estimation and Control*, Vol. 1, Academic Press, 1979.

[29] Bernstein, D. S., *Matrix Mathematics: Theory, Facts, and Formulas*, Princeton University Press, 2005.

[30] Julier, S. J. and Uhlmann, J. K., "New Extension of the Kalman Filter to Nonlinear Systems," *Proceedings of Signal Processing, Sensor Fusion, and Target Recognition VI*, 1997.

[31] Rehder, J., Gupta, K., Nuske, S., and Singh, S., "Global Pose Estimation with Limited GPS and Long Range Visual Odometry," *IEEE International Conference on Robotics and Automation*, 2012, pp. 627–633.

[32] Sibley, G., Sukhatme, G., and Matthies, L., "The Iterated Sigma Point Kalman Filter with Applications to Long Range Stereo," *Proceedings of Robotics: Science and Systems*, Philadelphia, USA, August 2006.

[33] Nister, D., Naroditsky, O., and Bergen, J., "Visual Odometry," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, June 2004, pp. I–652–I–659 Vol.1.

[34] Davison, A., Reid, I., Molton, N., and Stasse, O., "MonoSLAM: Real-Time Single Camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 6, June 2007, pp. 1052–1067.

[35] Konolige, K. and Agrawal, M., "FrameSLAM: From Bundle Adjustment to Real-Time Visual Mapping," *IEEE Transactions on Robotics*, Vol. 24, No. 5, Oct 2008, pp. 1066–1077.

[36] Strasdat, H., Montiel, J. M. M., and Davison, A. J., "Real-time Monocular SLAM: Why Filter?" *IEEE International Conference on Robotics and Automation*, May 2010, pp. 2657–2664.

[37] Steiner, T. J., *A Unified Vision and Inertial Navigation System for Planetary Hoppers*, Master's thesis,

Massachusetts Institute of Technology, 2012.

[38] Roumeliotis, S., Johnson, A., and Montgomery, J., "Augmenting inertial navigation with image-based motion estimation," *IEEE International Conference on Robotics and Automation*, Vol. 4, 2002, pp. 4326–4333.