2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Minimum Sum of Distances Estimator: Robustness and Stability

Yoav Sharon, John Wright, Yi Ma

***Abstract*— We consider the problem of estimating a state x from noisy and corrupted linear measurements $y = Ax + z + e$, where z is a dense vector of small-magnitude noise and e is a relatively sparse vector whose entries can be arbitrarily large. We study the behavior of the ℓ^1 estimator $\hat{x} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - A\boldsymbol{x}\|_1$, and analyze its breakdown point with respect to the number of corrupted measurements $\|e\|_0$. We show that the breakdown point is independent of the noise. We introduce a novel algorithm for computing the breakdown point for any given A, and provide a simple bound on the estimation error when the number of corrupted measurements is less than the breakdown point. As a motivational example we apply our algorithm to design a robust state estimator for an autonomous vehicle, and show how it can significantly improve performance over the Kalman filter.**

I. INTRODUCTION

The problem of estimating a state $\boldsymbol{x}_0 \in \mathbb{R}^n$ from $m > n$ noisy linear measurements $\boldsymbol{y} \approx A\boldsymbol{x}_0 \in \mathbb{R}^m$, arises in a vast number of applications. In some applications one can assume that the difference between $\boldsymbol{y}$ and $A\boldsymbol{x}_0$ is a small i.i.d. Gaussian noise $\boldsymbol{z} \in \mathbb{R}^m$:

$$\boldsymbol{y} = A\boldsymbol{x}_0 + \boldsymbol{z}. \tag{1}$$

In this case, the optimal estimate of $\boldsymbol{x}_0$ is the least-squares estimate: $\hat{\boldsymbol{x}}_2 = \left(A^T A\right)^{-1} A^T \boldsymbol{y} = \arg\min_x \|\boldsymbol{y} - A\boldsymbol{x}\|_2$. The least-square estimate is known as *stable* in the sense that the estimation error $\|\hat{\boldsymbol{x}}_2 - \boldsymbol{x}_0\|_2$ is bounded by a continuous function of $\boldsymbol{z}$. Thus, small noise causes only small estimation error. Often, however, some of the measurements in $\boldsymbol{y}$ can be corrupted by arbitrarily large errors. In this case, we instead must solve $\boldsymbol{x}_0$ from the equation

$$\boldsymbol{y} = A\boldsymbol{x}_0 + \boldsymbol{z} + \boldsymbol{e}, \tag{2}$$

where $\boldsymbol{e} \in \mathbb{R}^m$ has some arbitrarily large nonzero entries. One typical example is a GPS system, whose estimated position output can occasionally be considerably corrupted when the signals from the satellites are reflected off the surrounding terrain (i.e. multipath). Even one such corrupted measurement can cause arbitrarily large estimation error in the least-squares estimate.

When the state being estimated is a scalar ($n = 1$), the least-squares estimate $\hat{\boldsymbol{x}}_2$ is equivalent to taking a weighted average of the measurements. A known robust alternative to the average is the median. With the median, up to almost 50% of the measurements can be arbitrarily corrupted before the estimation error becomes unbounded. That is, the breakdown point of the median is 50%.

Taking the median, one essentially looks for the point which minimizes the sum of distances to all the measurements whereas taking the average minimizes the sum of the squares of these distances. One natural generalization of this concept to multivariate ($n > 1$) estimation[1] is to view the m measurements $\boldsymbol{y} \doteq [y_1, \ldots, y_m]^T$ as defining m hyperplanes:

$$H_i \doteq \left\{\boldsymbol{x} \in \mathbb{R}^n \,\middle|\, y_i = \boldsymbol{a}_i^T \boldsymbol{x}\right\},$$

where $\boldsymbol{a}_i^T \in \mathbb{R}^n$ is the corresponding row of the matrix $A \doteq [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m]^T$. Then the "median" estimate for $\boldsymbol{x}$ can be defined to be the point that minimizes the sum of distances to these hyperplanes:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \sum_{i=1}^{m} \left|y_i - \boldsymbol{a}_i^T \boldsymbol{x}\right| = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - A\boldsymbol{x}\|_1. \tag{3}$$

To understand why this estimate can be robust to errors, let us assume the noise is zero for now: $\boldsymbol{z} = \boldsymbol{0}$. That is, we try to solve $\boldsymbol{x}_0$ from the equation $\boldsymbol{y} = A\boldsymbol{x}_0 + \boldsymbol{e}$. If we could somehow compute $\boldsymbol{e}$, then $\boldsymbol{x}_0$ could be easily recovered from the clean system of equations $A\boldsymbol{x}_0 = \boldsymbol{y} - \boldsymbol{e}$. One approach to recovering $\boldsymbol{e}$ is to choose a matrix $B \in \mathbb{R}^{p \times m}$, $p = m - n$, with $BA = 0$, and define $\boldsymbol{w} = B\boldsymbol{y}$. Multiplying both sides of the measurement equation by B yields an underdetermined system of equations $\boldsymbol{w} = B\boldsymbol{e}$ in $\boldsymbol{e}$ alone. In the context of compressed sensing [2], it has recently been discovered that whenever $\boldsymbol{e}$ is sparse enough, it can be correctly recovered by solving the following ℓ^1-minimization problem:

$$\hat{\boldsymbol{e}} = \arg\min_{\boldsymbol{e}} \|\boldsymbol{e}\|_1 \quad \text{subject to} \quad \boldsymbol{w} = B\boldsymbol{e}. \tag{4}$$

So, in the noise free case, the two problems (3) and (4) are equivalent.

There is also a large literature analyzing the performance of (4) and related estimates in the presence of noise. The strongest available results ([3], [4], amongst others) have the following flavor: for some constants C and ρ, and almost all random matrices B, if one applies an ℓ^2-penalized version of (4) (i.e., the Lasso [5], [6]) and the number of errors $\|\boldsymbol{e}\|_0$ is less than $\rho \cdot n$, then the estimation error is bounded by

This work was supported by the following grants: NSF ECCS-0701676, NSF CRS-EHS-0509151, NSF CCF-TF-0514955, ONR YIP N00014-05-1-0633, and NSF IIS 07-03756. John Wright is also supported by a Microsoft Fellowship.

All authors are with the Coordinated Science Laboratory, Department of Electrical and Computer Engineering, the University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA. Email addresses: {ysharon2,jnwright,yima}@illinois.edu.

[1] Another multivariate generalization of the median occurs in robust center-point estimation, where the observations are themselves points (rather than inner products). There, the estimator that minimizes the sum of distances to the observations, known as the Fermat-Weber point, achieves a breakdown point of 50% [1, Theorem 2.2]. Although the estimator studied here also generalizes the median, it addresses the more general problem of robust linear regression.

978-1-4244-4524-0/09/$25.00 ©2009 AACC

$C \cdot \|z\|$ for some $C > 0$. However, specific forms of the constants C and ρ are difficult to derive. A similar bound can be derived when B is known to be a *restricted isometry* [3]. However, it requires prior knowledge of the noise level, and the estimation error depends on the number of corrupted measurements, with the bound C diverging to infinity when the error fraction ρ approaches the breakdown point. Similar results have also been obtained for greedy alternatives to ℓ^1-minimization [7]. In this setting, one does not require a bound on the noise term. However, it does require that the number of corrupted measurements be considerably lower than the breakdown point for ℓ^1-minimization.

Whereas most of the existing stability results and bounds are derived for the underdetermined case (4), in this paper, we directly study the stability of the ℓ^1 estimator for the overdetermined problem (3). Our bounds are weaker than those obtained in the asymptotic setting of large random matrices and small error fractions [4]. However, they hold for all matrices A, including the structured matrices arising in state estimation problems, and all error fractions ρ, up to the intrinsic breakdown point of the ℓ^1 estimator. Moreover, our bound has a very simple expression, whose derivation naturally suggests an algorithm for computing the intrinsic breakdown point of the ℓ^1 estimator. The complexity of our algorithm is exponentially lower than the existing alternative, and it is especially suitable for the kind of problems of interest for the system and control community – moderate-sized robust state estimation problems.

II. Preliminaries

Throughout, the 0-norm will denote the number of nonzero elements in a vector $\boldsymbol{v} \in \mathbb{R}^m$:

$$\|\boldsymbol{v}\|_0 \doteq \#\left\{i \middle| v_i \neq 0\right\}.$$

We will use $[m]$ to denote the set of indices $[m] \doteq \{1, 2, \ldots, m\}$. We will use the following notation for "positive" directional derivative of an arbitrary multivariate function $f : \mathbb{R}^m \to \mathbb{R}$:

$$D^+_{\boldsymbol{v}} f(\boldsymbol{x}) = \lim_{\varepsilon \searrow 0} \frac{f(\boldsymbol{x} + \varepsilon \boldsymbol{v}) - f(\boldsymbol{x})}{\varepsilon}.$$

Consider a general estimation problem, $\boldsymbol{y} = f(\boldsymbol{x}_0, \boldsymbol{z}, \boldsymbol{e})$, where $\boldsymbol{x}_0$ is the unknown state to be estimated, $\boldsymbol{z}$ is a noise term, $\boldsymbol{e}$ is a corruption term and $\boldsymbol{y}$ is the available measurements. Let $\hat{\boldsymbol{x}} = g(\boldsymbol{y})$ be some estimate. We say that for given $\boldsymbol{x}_0$ and $\boldsymbol{z}$, the estimate is robust up to T corrupted measurements (or T-robust) if there exists a smooth function $\beta(\boldsymbol{x}_0, \boldsymbol{z}) \in \mathbb{R}$ such that:

$$\forall \boldsymbol{e} : \text{if } \|\boldsymbol{e}\|_0 < T \text{ then } \|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2 \leq \beta(\boldsymbol{x}_0, \boldsymbol{z}). \tag{5}$$

The *breakdown point* of this scheme, $T^*(\boldsymbol{x}_0, \boldsymbol{z})$, is the minimum $T \in \mathbb{N}$ for which the estimation scheme is *not* T-robust. In other words,

$$T^*(\boldsymbol{x}_0, \boldsymbol{z}) \doteq \min\left\{T \in \mathbb{N} \,\middle|\, \sup_{\boldsymbol{e}, \|\boldsymbol{e}\|_0 \leq T} \|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2 = \infty\right\}.$$

We say T^* is a *stable* breakdown point if it does not depend on $\boldsymbol{x}_0$ and $\boldsymbol{z}$, i.e. $T^*(\boldsymbol{x}_0, \boldsymbol{z}) \equiv T^*$.

Throughout this paper we consider the problem of estimating $\boldsymbol{x}_0$ from $\boldsymbol{y}$:

$$\boldsymbol{y} = A\boldsymbol{x}_0 + \boldsymbol{z} + \boldsymbol{e},$$

where $\boldsymbol{x}_0 \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $\boldsymbol{z} \in \mathbb{R}^m$ and $\boldsymbol{e} \in \mathbb{R}^m$. For this problem we consider the Minimum Sum of Distances (MSoD) estimation scheme

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} C_{\boldsymbol{y}}(\boldsymbol{x}), \tag{6}$$

with the cost function

$$C_{\boldsymbol{y}}(\boldsymbol{x}) \doteq \|\boldsymbol{y} - A\boldsymbol{x}\|_1. \tag{7}$$

Our goal is to study whether the breakdown point of this estimate is stable and if so, how to compute it.

We start by giving results pertaining to the noiseless case, $\boldsymbol{z} = \boldsymbol{0}$. We assume T of the measurements can be corrupted. Geometrically, this means that the remaining $m - T$ measurement hyperplanes $H_i \doteq \left\{\boldsymbol{x} \in \mathbb{R}^n \mid y_i = \boldsymbol{a}_i^T \boldsymbol{x}\right\}$ pass through $\boldsymbol{x}_0$. We will let I denote the indices of these uncorrupted hyperplanes. The corrupted ones will be conveniently denoted by I^c. We ask whether these T hyperplanes can be positioned so that $\boldsymbol{x}_0$ no longer minimizes the cost function (7). Since $C_{\boldsymbol{y}}$ is convex, this will be true if and only if there exists a direction $\boldsymbol{v}$, from $\boldsymbol{x}_0$, along which the cost function does not increase, i.e. $D^+_{\boldsymbol{v}} C_{\boldsymbol{y}}(\boldsymbol{x}_0) \leq 0$. Since the uncorrupted hyperplanes pass through $\boldsymbol{x}_0$, moving in the direction of $\boldsymbol{v}$ from $\boldsymbol{x}_0$ will increase the distance to each of the uncorrupted hyperplanes at a rate of $\left|\boldsymbol{a}_i^T \boldsymbol{v}\right|$, $i \in I$. We have freedom in placing the corrupted hyperplanes, and so for each $\boldsymbol{v}$ we can position them so that moving in the direction of $\boldsymbol{v}$ will decrease the distance to each of the corrupted hyperplanes by a rate of $\left|\boldsymbol{a}_i^T \boldsymbol{v}\right|$, $i \in I^c$. In this case, which can be referred to as worst positioning of the corrupted hyperplanes given $\boldsymbol{v}$, the condition $D^+_{\boldsymbol{v}} C_{\boldsymbol{y}}(\boldsymbol{x}_0) \leq 0$ becomes

$$\sum_{i \in I} \left|\boldsymbol{a}_i^T \boldsymbol{v}\right| - \sum_{i \in I^c} \left|\boldsymbol{a}_i^T \boldsymbol{v}\right| \leq 0. \tag{8}$$

Because (8) represents the worst case for a given $\boldsymbol{v}$, $\boldsymbol{x}_0$ fails to minimize the cost function if and only if (8) holds for some $\boldsymbol{v}$. Thus we arrive at a lemma following the next definition:

Definition 2.1: $\widetilde{T}(A)$ is defined as the minimal integer T for which there exists $I \subset [m]$, $|I| = m - T$ and $\boldsymbol{v} \in \mathbb{R}^n$ such that (8) holds.

Lemma 2.1: Under the condition $\boldsymbol{z} = \boldsymbol{0}$, the breakdown point of the estimation scheme (6) is equal to $\widetilde{T}$ as defined in Definition 2.1, i.e. $T^*(\boldsymbol{x}_0, \boldsymbol{0}) = \widetilde{T}(A)$, $\forall \boldsymbol{x}_0 \in \mathbb{R}^n$.

In the next section we will consider the noisy case and show that this breakdown point is stable and the estimation error is bounded by a linear function of the noise magnitude $\|\boldsymbol{z}\|_2$ that does not depend on $\boldsymbol{x}_0$.

III. Proof of Robustness

We start with the following definition:

Definition 3.1: Given an arbitrary $T \in \mathbb{N}$ we call a set J' a *possibly extreme set* if there exists I, $I \supseteq J'$, $|I| = m - T$

such that the following holds:

$$\sum_{i\in J'\cup I^c} \left|\boldsymbol{a}_i^T \boldsymbol{\nu}_{J'}\right| \geq \sum_{i\in I\setminus J'} \left|\boldsymbol{a}_i^T \boldsymbol{\nu}_{J'}\right| \tag{9}$$

where $\boldsymbol{\nu}_{J'}$ is any of the singular vectors corresponding to the smallest singular value of the $|J'| \times n$ submatrix $A_{J'}$ of A containing those rows indexed by J': $\|A_{J'}\boldsymbol{\nu}_{J'}\|_2 = \sigma_{min}(A_{J'})\|\boldsymbol{\nu}_{J'}\|_2$ with $\sigma_{min}(\cdot)$ being the smallest singular value. We define Q_T to be the set of all possibly extreme sets for a given T.

The following is our main result:

Theorem 3.1: For any $T \in \{0, 1, \ldots, m\}$, if the number of corrupted measurements is not larger than T, then the estimation error is bounded as follows:

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2 \leq \left(\max_{J'\in Q_T} \frac{1}{\sigma_{min}(A_{J'})}\right) \|\boldsymbol{z}\|_2 . \tag{10}$$

Before proving Theorem 3.1 we emphasize a few observations. First note that if $T < \tilde{T}(A)$ then $\forall I \subset [m]$, $|I| = m - T$ the following holds:

$$\forall \boldsymbol{v} \in \mathbb{R}^n : \quad \sum_{i\in I} \left|\boldsymbol{a}_i^T \boldsymbol{v}\right| > \sum_{i\in I^c} \left|\boldsymbol{a}_i^T \boldsymbol{v}\right| . \tag{11}$$

Now, assume for some J' we have $\sigma_{min}(A_{J'}) = 0$. This implies that $\boldsymbol{a}_i^T \boldsymbol{\nu}_{J'} = 0 \ \forall i \in J'$. From (11) we see that in this case (9) can not hold and thus $J' \notin Q_T$. From this we conclude that $\sigma_{min}(A_{J'}) > 0 \ \forall J' \in Q_T$ and thus the expression inside the brackets in (10) must be finite. The fact that we have established a finite bound (when $\|\boldsymbol{z}\|_2$ is finite) for all $T < \tilde{T}(A)$, and $\tilde{T}(A)$ is independent of $\boldsymbol{x}_0$ and $\boldsymbol{z}$, proves that the breakdown point $T^*(\boldsymbol{x}_0, \boldsymbol{z}) \equiv \tilde{T}(A)$ is stable.

The second observation is that for $T' < T$ we have $Q_{T'} \subseteq Q_T$ and possibly even $Q_{T'} \subset Q_T$ where some of the smaller sets in Q_T may not be in $Q_{T'}$. Since $J' \subseteq J$ implies $\sigma_{min}(A_{J'}) \leq \sigma_{min}(A_J)$, losing the smaller sets from Q_T (as we reduce the number of corrupted measurements) can produce a smaller bound in Theorem 3.1

Definition 3.2: We define the following sets:

$$\begin{aligned} J_+(\boldsymbol{x}, \boldsymbol{y}) &\doteq \left\{i \in [m] \mid \boldsymbol{a}_i^T \boldsymbol{x} > y_i\right\} \\ J_0(\boldsymbol{x}, \boldsymbol{y}) &\doteq \left\{i \in [m] \mid \boldsymbol{a}_i^T \boldsymbol{x} = y_i\right\} \\ J_-(\boldsymbol{x}, \boldsymbol{y}) &\doteq \left\{i \in [m] \mid \boldsymbol{a}_i^T \boldsymbol{x} < y_i\right\} \end{aligned}$$

Also, for a point $\boldsymbol{x} \in \mathbb{R}^n$, $I_{\boldsymbol{x}}(\boldsymbol{y}) = J_0(\boldsymbol{x}, \boldsymbol{y}) \cap I$ is defined to be the set of uncorrupted hyperplanes passing through $\boldsymbol{x}$.

Proposition 3.2: For any $\hat{\boldsymbol{x}} \in \mathbb{R}^n$:

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2 \leq \frac{1}{\sigma_{min}\left(A_{I_{\hat{\boldsymbol{x}}}(\boldsymbol{y})}\right)} \|\boldsymbol{z}\|_2$$

Proof: Trivial since $\boldsymbol{z}_{I_{\hat{\boldsymbol{x}}}(\boldsymbol{y})} = A_{I_{\hat{\boldsymbol{x}}}(\boldsymbol{y})}(\hat{\boldsymbol{x}} - \boldsymbol{x}_0)$. ■

Our proof of Theorem 3.1 will go as follow. Assume $\boldsymbol{x}_0$, I, $\boldsymbol{z}$, $\boldsymbol{e}$ are given and let $\hat{\boldsymbol{x}}$ be the point minimizing the cost function. We will show that we can change only the noise and the corruption to $\boldsymbol{z}'$, $\boldsymbol{e}'$ such that $\|\boldsymbol{z}'\|_2 = \|\boldsymbol{z}\|_2$, $\boldsymbol{e}'_I = \boldsymbol{0}$, and the new corresponding minimizing point $\hat{\boldsymbol{x}}'$ achieves a larger estimation error, $\left\|\hat{\boldsymbol{x}}' - \boldsymbol{x}_0\right\|_2 \geq \|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2$. Furthermore, with the new $\boldsymbol{y}' = A\boldsymbol{x}_0 + \boldsymbol{z}' + \boldsymbol{e}'$ we will have $I_{\hat{\boldsymbol{x}}'}(\boldsymbol{y}') \in Q$. Applying then Proposition 3.2 on the new $\hat{\boldsymbol{x}}'$ and $\boldsymbol{y}'$, together with the fact that we did not decrease the estimation error, gives us (10). We will do this through several steps.

Proposition 3.3: Let $\boldsymbol{y}$ and the corresponding point $\hat{\boldsymbol{x}}$ which minimizes the cost function be given. For a different $\boldsymbol{y}'$, if there exists a point $\boldsymbol{x}'$ such that

$$\begin{aligned} J_+(\boldsymbol{x}', \boldsymbol{y}') &\subseteq J_+(\hat{\boldsymbol{x}}, \boldsymbol{y}) \\ J_-(\boldsymbol{x}', \boldsymbol{y}') &\subseteq J_-(\hat{\boldsymbol{x}}, \boldsymbol{y}) \\ J_0(\hat{\boldsymbol{x}}, \boldsymbol{y}) &\subseteq J_0(\boldsymbol{x}', \boldsymbol{y}') , \end{aligned} \tag{12}$$

then $\boldsymbol{x}'$ will minimize the cost function for $\boldsymbol{y}'$.

Proof: The rate of change of the cost function moving from $\boldsymbol{x}'$ in an arbitrary direction $\boldsymbol{v}$ is:

$$\begin{aligned} &D_{\boldsymbol{v}}^+ C_{\boldsymbol{y}'}(\boldsymbol{x}') = \\ &\sum_{i\in J_+(\boldsymbol{x}',\boldsymbol{y}')} \boldsymbol{a}_i^T \boldsymbol{v} + \sum_{i\in J_0(\boldsymbol{x}',\boldsymbol{y}')} \left|\boldsymbol{a}_i^T \boldsymbol{v}\right| - \sum_{i\in J_-(\boldsymbol{x}',\boldsymbol{y}')} \boldsymbol{a}_i^T \boldsymbol{v} \geq \\ &\sum_{i\in J_+(\hat{\boldsymbol{x}},\boldsymbol{y})} \boldsymbol{a}_i^T \boldsymbol{v} + \sum_{i\in J_0(\hat{\boldsymbol{x}},\boldsymbol{y})} \left|\boldsymbol{a}_i^T \boldsymbol{v}\right| - \sum_{i\in J_-(\hat{\boldsymbol{x}},\boldsymbol{y})} \boldsymbol{a}_i^T \boldsymbol{v} = \\ &D_{\boldsymbol{v}}^+ C_{\boldsymbol{y}}(\hat{\boldsymbol{x}}) > 0 \end{aligned}$$

■

Lemma 3.4: Assume $\boldsymbol{x}_0$, I, $\boldsymbol{z}$, $\boldsymbol{e}$ are given and let $\hat{\boldsymbol{x}}$ be the point minimizing the cost function. There exists $\boldsymbol{z}'$, $\boldsymbol{e}'$ such that $\|\boldsymbol{z}'\| = \|\boldsymbol{z}\|$, $\boldsymbol{e}'_I = \boldsymbol{0}$, and the new corresponding minimizing point $\hat{\boldsymbol{x}}'$ achieves a larger estimation error, $\left\|\hat{\boldsymbol{x}}' - \boldsymbol{x}_0\right\|_2 \geq \|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2$. Furthermore, either $\boldsymbol{v}' \doteq \hat{\boldsymbol{x}}' - \boldsymbol{x}_0 \propto \boldsymbol{\nu}_{I_{\hat{\boldsymbol{x}}'}(\boldsymbol{y}')}$ or $I_{\hat{\boldsymbol{x}}}(\boldsymbol{y}) \subsetneq I_{\hat{\boldsymbol{x}}'}(\boldsymbol{y}')$.

Proof: Define $\boldsymbol{v} \doteq \hat{\boldsymbol{x}} - \boldsymbol{x}_0$. If $\boldsymbol{v} \propto \boldsymbol{\nu}_{I_{\hat{x}}}$ then we are done. Otherwise set $\bar{\boldsymbol{v}} \propto \boldsymbol{\nu}_{I_{\hat{x}}}$, $\langle \bar{\boldsymbol{v}}, \boldsymbol{v}\rangle \geq 0$, $\|\bar{\boldsymbol{v}}\|_2 = 1$. Also, set $\bar{\boldsymbol{v}}^\perp$ to be the normalized vector perpendicular to $\bar{\boldsymbol{v}}$, in the span of $\boldsymbol{v}$ and $\bar{\boldsymbol{v}}$, and such that $\langle \bar{\boldsymbol{v}}^\perp, \boldsymbol{v}\rangle \geq 0$. Consider the vector function

$$f(\alpha) = \frac{\cos(\alpha)\bar{\boldsymbol{v}}^\perp + \sin(\alpha)\bar{\boldsymbol{v}}}{\left\|A_{I_{\hat{x}}(\boldsymbol{y})}\left(\cos(\alpha)\bar{\boldsymbol{v}}^\perp + \sin(\alpha)\bar{\boldsymbol{v}}\right)\right\|_2} \left\|\boldsymbol{z}_{I_{\hat{x}}(\boldsymbol{y})}\right\|_2 .$$

Define $\alpha_0 = \sin^{-1}(\langle \bar{\boldsymbol{v}}, \boldsymbol{v}\rangle) \in [0, \pi/2]$. Note that if we set

$$\begin{aligned} \bar{\boldsymbol{z}}_{I_{\hat{x}}(\boldsymbol{y})}(\alpha) &= A_{I_{\hat{x}}(\boldsymbol{y})} f(\alpha) \\ \bar{\boldsymbol{z}}_{[m]\setminus I_{\hat{x}}(\boldsymbol{y})}(\alpha) &= \boldsymbol{z}_{[m]\setminus I_{\hat{x}}(\boldsymbol{y})} \\ \bar{\boldsymbol{e}}_{I^c}(\alpha) &= \boldsymbol{e}_{I^c} + A_{I^c} f(\alpha) - A_{I^c} f(\alpha_0) \\ \bar{\boldsymbol{e}}_I(\alpha) &= \boldsymbol{0} \end{aligned}$$

then $\bar{\boldsymbol{z}}(\alpha_0) = \boldsymbol{z}$ and for $\alpha \in [\alpha_0, \pi/2]$ we have $\|\bar{\boldsymbol{z}}\|_2 = \|\boldsymbol{z}\|_2$.

We will set $\boldsymbol{z}' = \bar{\boldsymbol{z}}(\alpha^*)$, $\boldsymbol{e}' = \bar{\boldsymbol{e}}(\alpha^*)$ where
$\alpha^* = \max\{\pi/2, \tilde{\alpha}\}$

$$\tilde{\alpha} = \sup\left\{\alpha \;\middle|\; \begin{array}{c} \boldsymbol{a}_i f(\alpha) > z_i \,\forall i \in I \cap J_+(\hat{\boldsymbol{x}}, \boldsymbol{y}) \\ and \\ \boldsymbol{a}_i f(\alpha) < z_i \,\forall i \in I \cap J_-(\hat{\boldsymbol{x}}, \boldsymbol{y}) \end{array}\right\}.$$

With this choice of $\boldsymbol{z}'$ and $\boldsymbol{e}'$ we guarantee that (12) holds with $\boldsymbol{x}' = \boldsymbol{x}_0 + f(\alpha^*)$, and therefore $\boldsymbol{v}' = f(\alpha^*)$ is the new estimation error. If $\alpha^* = \pi/2$ then $\boldsymbol{v}' \propto \boldsymbol{\nu}_{I_{\hat{x}'}}$. Otherwise one of the strict inequalities in (13) must become an inequality with α^*, which implies $I_{\hat{x}}(\boldsymbol{y}) \subsetneq I_{\hat{x}'}(\boldsymbol{y}')$. To complete the proof we are left to show that

$$\|f(\alpha)\|_2 = \frac{\left\|\cos(\alpha)\bar{\boldsymbol{v}}^{\perp} + \sin(\alpha)\bar{\boldsymbol{v}}\right\|_2}{\left\|A_{I_{\hat{x}}(\boldsymbol{y})}\left(\cos(\alpha)\bar{\boldsymbol{v}}^{\perp} + \sin(\alpha)\bar{\boldsymbol{v}}\right)\right\|_2}\left\|\boldsymbol{z}_{I_{\hat{x}}(\boldsymbol{y})}\right\|_2 \tag{13}$$

is monotonically non-decreasing.

The numerator in (13) as well as the $\left\|\boldsymbol{z}_{I_{\hat{x}}(\boldsymbol{y})}\right\|_2$ term are constants. Because the singular vector $\bar{\boldsymbol{v}}$ is an eigenvector of $A^T_{I_{\hat{x}}(\boldsymbol{y})}A_{I_{\hat{x}}(\boldsymbol{y})}$ we have that $\left\langle A_{I_{\hat{x}}(\boldsymbol{y})}\bar{\boldsymbol{v}}^{\perp}, A_{I_{\hat{x}}(\boldsymbol{y})}\bar{\boldsymbol{v}}\right\rangle = 0$, thus the derivative of the denominator with respect to α is

$$\frac{\left(-\left\|A_{I_{\hat{x}}(\boldsymbol{y})}\bar{\boldsymbol{v}}^{\perp}\right\|_2^2 + \left\|A_{I_{\hat{x}}(\boldsymbol{y})}\bar{\boldsymbol{v}}\right\|_2^2\right)\sin(\alpha)\cos(\alpha)}{\left\|A_{I_{\hat{x}}(\boldsymbol{y})}\left(\cos(\alpha)\bar{\boldsymbol{v}}^{\perp} + \sin(\alpha)\bar{\boldsymbol{v}}\right)\right\|_2}.$$

This is always non-positive because $\alpha \in [0, \pi/2]$ and $\bar{\boldsymbol{v}}$ is the singular vector corresponding to the smallest singular value. ∎

By iterating the procedure described in the last lemma, each time adding at least one more element to $I_{\hat{x}'}(\boldsymbol{y}')$, we arrive at the following Corollary:

Corollary 3.5: Assume $\boldsymbol{x}_0$, I, $\boldsymbol{z}$, $\boldsymbol{e}$ are given and let $\hat{\boldsymbol{x}}$ be the point minimizing the cost function. There exists $\boldsymbol{z}'$, $\boldsymbol{e}'$ such that $\|\boldsymbol{z}'\|_2 = \|\boldsymbol{z}\|_2$, $\boldsymbol{e}'_I = \boldsymbol{0}$, the new corresponding minimizing point $\hat{\boldsymbol{x}}'$ achieves a larger estimation error, $\left\|\hat{\boldsymbol{x}}' - \boldsymbol{x}_0\right\|_2 \geq \|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2$, and $\boldsymbol{v}' \doteq \hat{\boldsymbol{x}}' - \boldsymbol{x}_0 \propto \boldsymbol{\nu}_{I_{\hat{x}'}}$.

Remark 3.1: Without loss of generality, for a given $\boldsymbol{y} \in \mathbb{R}^m$ and an arbitrary direction $\boldsymbol{v} \in \mathbb{R}^n$, we can assume that $\boldsymbol{a}_i^T\boldsymbol{v} \geq 0\ \forall i \in [m]$. This is because we can arbitrarily negate some of the $\boldsymbol{a}_i$'s and their corresponding y_i's without affecting the cost function (7).

Lemma 3.6: Assume $\boldsymbol{x}_0$, I, $\boldsymbol{z}$, $\boldsymbol{e}$ are given. Let $\hat{\boldsymbol{x}}$ be the point minimizing the cost function and assume $\boldsymbol{v} \doteq \hat{\boldsymbol{x}} - \boldsymbol{x}_0 \propto \boldsymbol{\nu}_{I_{\hat{x}}(\boldsymbol{y})}$. If $I_{\hat{x}}(\boldsymbol{y}) \notin Q$ then there exists $\boldsymbol{z}'$, $\boldsymbol{e}'$ such that $\|\boldsymbol{z}'\|_2 = \|\boldsymbol{z}\|_2$, $\boldsymbol{e}'_I = \boldsymbol{0}$, the new corresponding minimizing point $\hat{\boldsymbol{x}}'$ achieves a larger estimation error, $\left\|\hat{\boldsymbol{x}}' - \boldsymbol{x}_0\right\|_2 \geq \|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2$, and $I_{\hat{x}}(\boldsymbol{y}) \subsetneq I_{\hat{x}'}(\boldsymbol{y}')$.

Proof: WLOG (see Remark 3.1) assume $\boldsymbol{a}_i^T\boldsymbol{v} \geq 0\ \forall i \in [m]$. The rate of change going in direction $-\boldsymbol{v}$ from $\hat{\boldsymbol{x}}$ is:

$$-\sum_{i \in J_+(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| + \sum_{i \in J_0(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| + \sum_{i \in J_-(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right|. \tag{14}$$

Because $\hat{\boldsymbol{x}}$ minimizes the cost function, (14) must be non-negative. If indeed $I_{\hat{x}}(\boldsymbol{y}) \notin Q$ then from the fact that (9) is not satisfied for $J' = I_{\hat{x}}(\boldsymbol{y})$ we have

$$\sum_{i \in I \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| > \sum_{i \in I_{\hat{x}}(\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| + \sum_{i \in I^c}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| - \sum_{i \in I \cap J_+(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right|.$$

Now given that (14) is nonnegative we can write

$$\begin{aligned}&\sum_{i \in I_{\hat{x}}(\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| - \sum_{i \in I \cap J_+(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| \geq \\ &\sum_{i \in I^c \cap J_+(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| - \sum_{i \in I^c \cap J_0(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| - \sum_{i \in I^c \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| - \sum_{i \in I \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right|.\end{aligned}$$

Combining these last two inequalities we get

$$2\sum_{i \in I \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| > 2\sum_{i \in I^c \cap J_+(\hat{\boldsymbol{x}},\boldsymbol{y})}\left|\boldsymbol{a}_i^T\boldsymbol{v}\right| \geq 0$$

which implies that $I \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})$ cannot be empty. Now, for every z_i, $i \in I \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})$, we have

$$z_i = y_i - \boldsymbol{a}_i^T\boldsymbol{x}_0 = y_i + \boldsymbol{a}_i^T\boldsymbol{v} - \boldsymbol{a}_i^T\hat{\boldsymbol{x}} > 0.$$

Arbitrarily choose $i' \in I \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})$ and consider the following:

$$\begin{aligned}\bar{\boldsymbol{z}}_{I_{\hat{x}}(\boldsymbol{y})}(\alpha) &= A_{I_{\hat{x}}(\boldsymbol{y})}(1+\alpha)\boldsymbol{v} \\ \bar{z}_{i'}(\alpha) &= \sqrt{z_{i'}^2 + \left(1 - (1+\alpha)^2\right)\left\|\boldsymbol{z}_{I_{\hat{x}}(\boldsymbol{y})}\right\|_2^2} \\ \bar{\boldsymbol{z}}_{[m]\setminus(I_{\hat{x}}(\boldsymbol{y})\cup\{i'\})}(\alpha) &= \bar{\boldsymbol{z}}_{[m]\setminus(I_{\hat{x}}(\boldsymbol{y})\cup\{i'\})} \\ \bar{\boldsymbol{e}}_{I^c}(\alpha) &= \bar{\boldsymbol{e}}_{I^c}(\alpha) + \alpha A_{I^c}\boldsymbol{v} \\ \bar{\boldsymbol{e}}_I &= \boldsymbol{0}.\end{aligned}$$

with $\alpha \geq 0$. Note that $\|\bar{\boldsymbol{z}}(\alpha)\|_2$ is constant, and $\bar{\boldsymbol{z}}(0) = \boldsymbol{z}$. We will set $\boldsymbol{z}' = \boldsymbol{z}(\alpha^*)$ and $\boldsymbol{e}' = \bar{\boldsymbol{e}}(\alpha^*)$ where

$$\alpha^* = \sup\left\{\alpha\left|\boldsymbol{a}_i^T(1+\alpha)\boldsymbol{v} < \bar{z}_i(\alpha)\ \forall i \in I \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})\right.\right\}.$$

For every $\alpha \in [0, \alpha^*)$ we have that (12) holds with $\boldsymbol{x}' = \boldsymbol{x}_0 + (1+\alpha)\boldsymbol{v}$ and therefore $(1+\alpha)\boldsymbol{v}$ is the new estimation error. With $\alpha = \alpha^*$ we also have $\boldsymbol{a}_i^T(1+\alpha)\boldsymbol{v} = z'_i \Leftrightarrow \boldsymbol{a}_i^T\boldsymbol{x}' = y'_i$ for some $i \in I \cap J_-(\hat{\boldsymbol{x}},\boldsymbol{y})$. This implies $I_{\hat{x}}(\boldsymbol{y}) \subsetneq I_{\hat{x}'}(\boldsymbol{y}')$. ∎

By iterating the procedures described in (3.4) and (3.6) several times as necessary we arrive at the final corollary:

Corollary 3.7: Assume $\boldsymbol{x}_0$, I, $\boldsymbol{z}$, $\boldsymbol{e}$ are given and let $\hat{\boldsymbol{x}}$ be the point minimizing the cost function. There exists $\boldsymbol{z}'$, $\boldsymbol{e}'$ such that $\|\boldsymbol{z}'\|_2 = \|\boldsymbol{z}\|_2$, $\boldsymbol{e}'_I = \boldsymbol{0}$, and the new corresponding minimizing point $\hat{\boldsymbol{x}}'$ achieves a larger estimation error, $\left\|\hat{\boldsymbol{x}}' - \boldsymbol{x}_0\right\|_2 \geq \|\hat{\boldsymbol{x}} - \boldsymbol{x}_0\|_2$. Furthermore, $I_{\hat{x}'}(\boldsymbol{y}') \in Q$.

The last Corollary, together with Proposition 3.2, proves Theorem 3.1.

IV. Computing the Breakdown Point

Definition 2.1 does not immediately suggest an algorithm for computing $\widetilde{T} = T^*$, because it requires checking condition (8) for all $\boldsymbol{v} \in \mathbb{R}^n$, $\|\boldsymbol{v}\|_2 = 1$, and there are infinitely many such $\boldsymbol{v}$. The following Lemma 4.1, however, states that it is sufficient to check only a finite subset of $\mathbb{R}^n$:

Lemma 4.1: Condition (8) holds for some $I \subset [m]$ and $\boldsymbol{v} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$ if and only if there exist $J \subset [m]$ and $\boldsymbol{v}' \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$ with the following properties: $|J| = n-1$; $\{\boldsymbol{a}_i\}_{i \in J}$ is a set of $n-1$ linearly independent vectors; $\boldsymbol{a}_i^T\boldsymbol{v}' = 0$ $\forall i \in J$; and (8) holds for $\boldsymbol{v}'$.

The *if* direction in 4.1 is trivial. In the degenerate case where $\dim\operatorname{span}\{\boldsymbol{a}_i\}_{i \in I} \leq n-1$ the *only if* is also trivial since (8) will hold for any nonzero vector which is not in the span of $\{\boldsymbol{a}_i\}_{i \in I}$. The *only if* direction in the non-degenerate case is an immediate corollary of the following proposition:

Proposition 4.2: Assume I and $\boldsymbol{v}$ are given, and $\dim\operatorname{span}\{\boldsymbol{a}_i\}_{i \in I} = n$. Define $J(\boldsymbol{v}) \doteq \{i \in I\,|\boldsymbol{a}_i\boldsymbol{v} = 0\}$ and $d(J) \doteq \dim\operatorname{span}\{\boldsymbol{a}_i\}_{i \in J}$. If Condition (8) holds for $\boldsymbol{v} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$ and $d(J(\boldsymbol{v})) < n-1$ then there exists $\boldsymbol{v}' \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$ for which (8) also holds but in addition $d(J(\boldsymbol{v}')) > d(J(\boldsymbol{v}))$.

Proof: WLOG we can assume $\boldsymbol{a}_i^T\boldsymbol{v} \geq 0\ \forall i \in [m]$. Consider the following set of equations in $\boldsymbol{z} \in \mathbb{R}^n$:

$$\sum_{i\in I^c} \boldsymbol{a}_i^T\boldsymbol{z} = 0 \tag{15}$$

$$\boldsymbol{a}_i^T\boldsymbol{z} = 0 \quad \forall i \in J(\boldsymbol{v}). \tag{16}$$

In the case that $d(J(\boldsymbol{v})) = \dim \operatorname{span}\{\boldsymbol{a}_i\}_{i\in J(\boldsymbol{v})} < n-1$, there is a nontrivial solution $\tilde{\boldsymbol{z}} \neq 0$ to (15) and (16). By changing the sign of $\tilde{\boldsymbol{z}}$ if necessary, we can assume

$$\sum_{i\in I} \boldsymbol{a}_i^T\tilde{\boldsymbol{z}} \leq 0. \tag{17}$$

Define the set $P \doteq \left\{i \in I \,\middle|\, \boldsymbol{a}_i^T\tilde{\boldsymbol{z}} < 0\right\}$ and $\alpha \doteq \min_{i\in P} \frac{\boldsymbol{a}_i^T\boldsymbol{v}}{-\boldsymbol{a}_i^T\tilde{\boldsymbol{z}}}$. Note that from (17) and the assumption that $d(I) = n$, P cannot be empty and thus α is well defined and positive. Also note that P contains only the indices of vectors from I which are linearly independent of $\{\boldsymbol{a}_i\}_{i\in J(\boldsymbol{v})}$. Set $\boldsymbol{v}' = \boldsymbol{v} + \alpha\tilde{\boldsymbol{z}}$. By our choice of α we have for some $i' \in P \subset I \setminus J$ that $\boldsymbol{a}_{i'}^T\boldsymbol{v}' = 0$. Since $\tilde{\boldsymbol{z}}$ satisfies (16) this gives us $J(\boldsymbol{v}') \supsetneq J(\boldsymbol{v})$ and $d(J(\boldsymbol{v}')) > d(J(\boldsymbol{v}))$. From (15) we have

$$\sum_{i\in I^c} \left|\boldsymbol{a}_i^T\boldsymbol{v}'\right| \geq \sum_{i\in I^c} \boldsymbol{a}_i^T(\boldsymbol{v} + \alpha\tilde{\boldsymbol{z}}) = \sum_{i\in I^c} \boldsymbol{a}_i^T\boldsymbol{v} = \sum_{i\in I^c} \left|\boldsymbol{a}_i^T\boldsymbol{v}\right|. \tag{18}$$

By our choice of α we also have $\boldsymbol{a}_i^T\boldsymbol{v}' \geq 0\ \forall i \in I$. Together with (17) this gives us

$$\begin{aligned}\sum_{i\in I} \left|\boldsymbol{a}_i^T\boldsymbol{v}'\right| &= \sum_{i\in I} \boldsymbol{a}_i^T\boldsymbol{v}' = \sum_{i\in I} \boldsymbol{a}_i^T\boldsymbol{v} + \alpha\sum_{i\in I} \boldsymbol{a}_i^T\tilde{\boldsymbol{z}} \\ &\leq \sum_{i\in I} \left|\boldsymbol{a}_i^T\boldsymbol{v}\right|. \end{aligned} \tag{19}$$

Combining (18), (19) and the fact that (8) holds for $\boldsymbol{v}$ implies that (8) also holds for $\boldsymbol{v}'$. ■

Given $J \subset I$, $|J| = n-1$, $d(J) = n-1$, the condition $A_J\boldsymbol{v}' = \boldsymbol{0}$ determines $\boldsymbol{v}'$ uniquely up to scale. The validity of condition (8) is unchanged by scaling $\boldsymbol{v}'$. Thus, we could equivalently define $T^*(A)$ to be the minimal integer T such that there exists $J \subset [m]$ of size $|J| = n-1$, $d(J) = n-1$, and $I \subset [m]$ of size $|I| = m - T$ for which condition (8) holds for $\boldsymbol{v}'$ satisfying $A_J\boldsymbol{v}' = \boldsymbol{0}$. Fix J (and a corresponding $\boldsymbol{v}$), and sort the $|\boldsymbol{a}_i^T\boldsymbol{v}|$ such that $\left|\boldsymbol{a}_{r_1}^T\boldsymbol{v}\right| \geq \left|\boldsymbol{a}_{r_2}^T\boldsymbol{v}\right| \geq \ldots \geq \left|\boldsymbol{a}_{r_m}^T\boldsymbol{v}\right|$. Then, condition (8) holds for some I of size $m-T$ if and only if it holds for $I \doteq \{r_{T+1}\ldots r_m\}$. We can therefore compute $T^*(A)$ by checking this condition for every subset J of size $n-1$. This idea is formalized as Algorithm 1.

The computation time of Algorithm 1 is

$$\binom{m}{n}\left(t_{sle}(n-1) + t_{mv}(m) + t_{sort}(m)\right), \tag{20}$$

where $t_{sle}(n) = \mathrm{O}\left(n^3\right)$, $t_{mv}(n) = \mathrm{O}\left(n^2\right)$ and $t_{sort}(n) = \mathrm{O}(n\log n)$ are the times it takes to solve a system of linear equations, to compute a matrix-vector multiplication, and to sort, respectively. When both m and n grow, $\binom{m}{n}$, and thus the computation time of our algorithm, grows exponentially. In many control applications, however, the number of variables describing the state of the system, n, is fixed, while the number of measurements, m, is flexible. In this case, where n is fixed, our algorithm's computation time is polynomial in m. We further note, that while the running time of the algorithm might still be relatively large in practice, from the engineering design point of view it needs to be executed only once during the design of the system to analyze its performance. In real-time only (6) needs to be evaluated, which can be done very efficiently using linear programming.

Algorithm 1 Computing $T^*(A)$

Input: $A \in \mathbb{R}^{m\times n}$.

- Set $T \leftarrow m$ and let $J_1,\ldots,J_N$, $N = \binom{m}{n-1}$, be all the subsets of $[m] \doteq \{1\ldots m\}$ containing $n-1$ indices.
- **for** $k = 1 : N$ **do**
  - **if** $\dim\operatorname{span}\{\boldsymbol{a}_i\}_{i\in J_k} = n-1$ **then**
    - Find a nontrivial solution $\boldsymbol{v} \in \mathbb{R}^n$ such that $\boldsymbol{a}_i^T\boldsymbol{v} = 0\ \forall i \in J_k$.
    - Find the order $r_1 \ldots r_m$ such that $\left|\boldsymbol{a}_{r_1}^T\boldsymbol{v}\right| \geq \left|\boldsymbol{a}_{r_2}^T\boldsymbol{v}\right| \geq \ldots \geq \left|\boldsymbol{a}_{r_m}^T\boldsymbol{v}\right|$.
    - Find the smallest integer, s, such that $\sum_{i=1}^{s}\left|\boldsymbol{a}_{r_i}^T\boldsymbol{v}\right| \geq \sum_{i=s+1}^{m}\left|\boldsymbol{a}_{r_i}^T\boldsymbol{v}\right|$.
    - Set $T \leftarrow \min\{T, s\}$.
  - **end if**
- **end for**

Output: T.

The algorithm described above is different from the existing algorithm in the literature for computing the breakdown point. In the introduction we have mentioned that in the absence of noise, (3) and (4) are equivalent problems when $B \in \mathbb{R}^{p\times m}$, $p = m-n$, $BA = 0$. The following result, proved in [8] and in [2, §II], states that the ability of (4) to recover $\boldsymbol{e}$ from the underdetermined linear system $\boldsymbol{w} = B\boldsymbol{e}$ depends only on the sign pattern of $\boldsymbol{e}$:

Theorem 4.3: If for some $\boldsymbol{e}' \in \mathbb{R}^n$, we have

$$\boldsymbol{e}' = \arg\min_{\boldsymbol{e}} \|\boldsymbol{e}\|_1 \quad \text{subject to} \quad B\boldsymbol{e} = B\boldsymbol{e}', \tag{21}$$

then for all $\tilde{\boldsymbol{e}}$ such that $\operatorname{sign}(\tilde{e}_i) = \operatorname{sign}(e_i')$, $i = 1\ldots n$,

$$\tilde{\boldsymbol{e}} = \arg\min_{\boldsymbol{e}} \|\boldsymbol{e}\|_1 \quad \text{subject to} \quad B\boldsymbol{e} = B\tilde{\boldsymbol{e}}.$$

From this result, to determine whether we can recover any T-sparse signal $\boldsymbol{e}$ (i.e. $\|\boldsymbol{e}\|_0 = T$), we only need to check one $\boldsymbol{e}$ for each T-sparse sign pattern. Specifically:

$$T^* = \min\left\{T \in \mathbb{N} \,\middle|\, \exists \boldsymbol{e}' \in E_T : \boldsymbol{e}' \neq \arg\min_{\boldsymbol{e}|B\boldsymbol{e}=B\boldsymbol{e}'} \|\boldsymbol{e}\|_1\right\} \tag{22}$$

where $E_T \doteq \{\boldsymbol{e} \in \mathbb{R}^m \mid \forall i : e_i \in \{-1,0,1\},\ \|\boldsymbol{e}\|_0 = T\}$.

Since $|E_T| = 2^T\binom{m}{T}$, a straightforward algorithm for computing (22) requires time

$$\sum_{T=1}^{T^*} 2^T\binom{m}{T} t_{lp}(m \times p), \tag{23}$$

where t_{lp} is the time it takes to solve the linear programming problem (21). We note that instead of actually solving for the

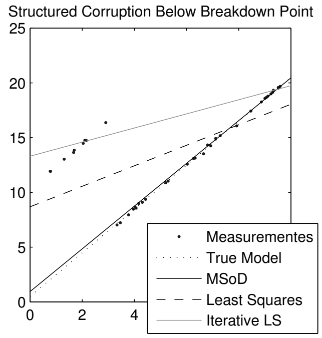

Fig. 1. We attempt to estimate a line model from which 40 noisy and corrupted points are drawn. The breakdown point of the MSoD estimator is 10 points. Corrupting the 10 leftmost points corresponds to the worst-case in which the MSoD will fail. In the example shown here we corrupted only the 9 leftmost points. Shown in the plot are the initial model estimated using least-squares for all the points, the model estimated by the iterative least-squares method, and that estimated by the MSoD. We can see that the MSoD works well, but the iterative trimming method, labeled "iterative LS," fails to converge to a good model.

right hand side of (21), one can check if e' minimizes the right hand side by looking for appropriate sub-gradients (see [2, §II]). This alternative approach, however, still requires solving a linear programming problem of similar size.

It is easy to see that the running time of our algorithm (20) is exponentially faster than the alternative (23) when n/m is small compared to T^*/m (i.e. A is very tall) or when n/m is very close to one (i.e. A is almost square). The first case is precisely the interest of robust estimation – the number of measurements needs to be large so as to tolerate more errors. This is the case for the robust state estimation problem one often encounters in control systems.

V. Comparison to other Robust Estimators

In this section we compare the Minimum Sum of Distances (MSoD) estimator to other typical robust estimation schemes in the literature.

A. Iterative Trimming

Arguably, this is the simplest robust estimator. Its application involves calculating an estimate using all (noisy and corrupted) measurements, say by least squares in our case. After discarding a certain number of measurements which are most inconsistent with the estimate, one recomputes the estimate using the remaining measurements. One may iterate the above process until only a predefined number of measurements remains, or until the residual error of the remaining measurements drops below some predefined level.

The main drawback of this method is that for certain corruption, the initial estimate from all the data can be made to favor some of the corrupted measurements over the uncorrupted measurements. We are not aware of any work that carefully analyzes the breakdown point of such an iterative method. However, we found that we can make this method fail using far fewer corrupted measurements than the breakdown point calculated for the MSoD estimator. Figure 1 shows a simple example in which the iterative least squares method fails but MSoD succeeds.

B. Random Sampling

Another popular approach to obtain robust estimate is through the RANdom SAmpling Consensus (RANSAC) method [9]. In our context, this corresponds to randomly selecting n of the m measurements (equations) and solving $\boldsymbol{x}$. One then checks how many other measurements are consistent with this estimate, say error incurred is below some level. The algorithm repeatedly select sets of n measurements until an estimate with high consensus is obtained. In theory, this approach has a breakdown point of 50%.

With p randomly selected sets of n measurements, the probably that at least one set contains no corrupted measurements at all is $1-(1-q^n)^p$ where q is the percentage of uncorrupted points. When n is small, this probability of success can be very high with relatively small number of selections – the reason why RANSAC has been very popular amongst practitioners. However, ensuring a fixed probability of success requires that the number of selections p grows *exponentially* in n, making it utterly inefficient when the dimension n is high. Linear programming solvers which minimize the MSoD cost function, on the other hand, require time polynomial in the size of the matrix A. Hence, MSoD is more scalable than RANSAC in dimension n, despite a lower breakdown point.[2]

VI. Application - Vehicle Position Estimation

In this subsection we present a "real-life" application that demonstrates the potential benefits of the Minimum Sum of Distances Estimator (MSoD). The problem which we address is estimating the position, orientation and velocity of a vehicle moving in 2D. The vehicle has inertial navigation sensors (gyroscopes) that generate noisy measurements of its velocity v and its rate of orientation change $\dot{\theta}$. In addition, the vehicle receives noisy measurements of its east, e, and north, n, position. A typical source for such measurements is a GPS system, which may produce corrupted or erroneous measurements due to multi-paths. The inertial measurements are generated every t_s seconds, while the position measurements are generated every T_s seconds, with $t_s \ll T_s$.

Given the car state at time t_0, its position at time t_1 is

$$\begin{aligned} e(t_1) &= e(t_0) + \textstyle\int_{t_0}^{t_1} \cos\theta(\tau)\, v(\tau)\, d\tau \\ n(t_1) &= n(t_0) + \textstyle\int_{t_0}^{t_1} \sin\theta(\tau)\, v(\tau)\, d\tau. \end{aligned}$$

Denote by $\hat{\cdot}$ our estimate of the car state and by $\boldsymbol{x} = \left(e-\hat{e}, n-\hat{n}, \theta-\hat{\theta}, v-\hat{v}\right)^T$ our (presumably small) estimation error. Denote by g_e, g_n the position measurements and by $\boldsymbol{y}(t) \doteq \left(y_0^T(t), \ldots, y_d(t)^T\right)^T$ the measurement residuals over a dT_s-time period, where

$$y_k(t) \doteq \begin{pmatrix} g_e(t+kT_S) - \hat{e}(t+kT_S) \\ g_n(t+kT_S) - \hat{n}(t+kT_S) \end{pmatrix} \approx \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \boldsymbol{x}(t+kT_s) \doteq C\boldsymbol{x}_k(t).$$

[2]It has been shown in the literature that for randomly generated A, the breakdown point of MSoD grows linearly in m [2], [10]. However, the fraction is normally bounded from above by 1/3.

Based on our assumptions, we can write

$$\begin{aligned}&\boldsymbol{x}\left(t+T_s\right) \approx \\ &\boldsymbol{x}\left(t\right) + \begin{pmatrix} \int_t^{t+T_s} \cos\theta(\tau)v(\tau)d\tau - \int_t^{t+T_s} \cos\hat{\theta}(\tau)\hat{v}(\tau)d\tau \\ \int_t^{t+T_s} \sin\theta(\tau)v(\tau)d\tau - \int_t^{t+T_s} \sin\hat{\theta}(\tau)\hat{v}(\tau)d\tau \\ 0 \\ 0 \end{pmatrix} \approx \\ &\begin{pmatrix} 1 & 0 & \int_t^{t+T_s} -\sin\theta(\tau)v(\tau)d\tau & \int_t^{t+T_s} \cos\theta(\tau)d\tau \\ 0 & 1 & \int_t^{t+T_s} \cos\theta(\tau)v(\tau)d\tau & \int_t^{t+T_s} \sin\theta(\tau)d\tau \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \boldsymbol{x}\left(t\right) \doteq \\ &F\left(t\right)\boldsymbol{x}\left(t\right)\end{aligned}$$

so that

$$\boldsymbol{y}\left(t\right) \approx \begin{pmatrix} C \\ CF(t) \\ \vdots \\ CF(t+(d-1)T_s)\ldots F(t+T_s)F(t) \end{pmatrix} \boldsymbol{x}\left(t\right) \doteq A\left(t\right)\boldsymbol{x}\left(t\right). \tag{24}$$

The approximations are due to the linearization of the nonlinear relation between the presumably small estimation error and the measurement residuals, and due to the noise and corruptions of the measurements.

Equation (24) is the linear model on which we apply our estimation scheme. Every time a new position measurement is generated we use it together with the last d position measurements to correct the vehicle estimated state. The matrix $A\left(t\right)$ and the estimated expected positions in the $\boldsymbol{y}$ vector are regenerated every time a new position measurement arrives to reflect our best estimate so far.

Simulation results are given in Figure 2. In this simulation, the breakdown point, calculated by Algorithm 1, ranges from 4 to 6, depending on the vehicle maneuvers. While the number of corrupted measurements occasionally exceeded the breakdown point, the results were still remarkably good. This is because the breakdown point represent a worst case scenario whose probability is relatively low. For comparison we also show in Figure 2 simulation results when a standard nonlinear Kalman filter was used for this system.

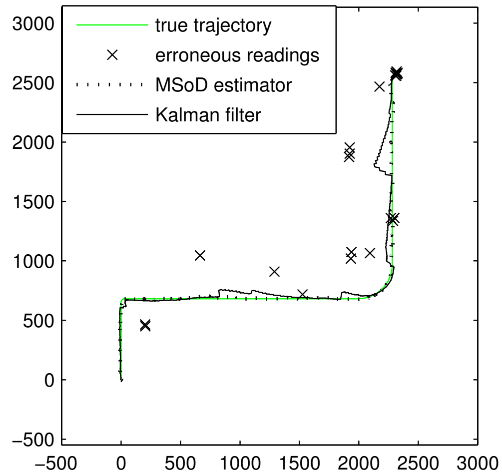

Fig. 2. Estimating a vehicle position which is moving in a 2D plane from noisy and corrupted measurements. The MSoD estimation scheme was applied on the linear model (24) using $d = 19$. The units in the plot are meters. The car average velocity is $85_{km/h}$. New position measurements are generated every $T_s = 1$ seconds. Uncorrupted position measurements have noise with 10_m standard deviation. Corrupted measurements have errors which are uniformly distributed up to 400_m. The system have a 0.06 (6%) probability of switching from an uncorrupted to a corrupted mode, and a 0.5 probability of switching from a corrupted mode to an uncorrupted mode. The maximum and the average magnitude of the position errors were 55_m and 9_m, respectively. For comparison we also show the results of using standard nonlinear Kalman filter. The standard deviation of the position errors, used to calculate the Kalman gains, was 200_m. The maximum and the average magnitude of the position errors were 157_m and 30_m, respectively.

VII. Conclusion

The main contribution of this paper was to show that the MSoD estimator, which was known to be *robust* with respect to corruption, is also *stable* with respect to noise. We also showed how to quantify the robustness and stability properties for deterministic matrices. Further study, for which the results in this paper can be used as a basis, is still needed. Key problems include developing a probabilistic or average-case analysis, as well as studying whether reweighting (by scaling the $\boldsymbol{a}_i$'s) can improve the estimator.

Acknowledgment

The authors would like to thank the following people for stimulating discussions on the problems studied in this paper: Professor Harm Derksen from the Mathematics Department at the University of Michigan; Professor Mike Wakin of the Colorado School of Mines; Professor Robert Fossum from the Mathematics Department, Shankar Rao and Arvind Ganesh from the Department of Electrical and Computer Engineering and Nir Friedman from the Physics Department, all four at the University of Illinois at Urbana-Champaign.

References

[1] H. P. Lopuha and P. J. Rousseeuw, "Breakdown points of affine equivariant estimators of multivariate location and covariance matrices," *Annals of Statistics*, vol. 19, no. 1, pp. 229–248, 1991.

[2] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[3] E. Candes and P. A. Randall, "Highly robust error correction by convex programming," *preprint, http://arxiv.org/abs/cs.IT/0612124*, 2006.

[4] D. Donoho, "For most large underdetermined systems of linear equations, the minimal ℓ^1-norm near-solution approximates the sparsest near-solution," *Communication on Pure and Applied Mathematics*, vol. 59, no. 7, pp. 907–934, March 2006.

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

[6] N. Meinshausen and B. Yu, "Lasso-type recovery of sparse representations for high-dimensional data," *to appear in Annals of Statistics, preprint at http://arxiv.org/abs/0806.0145*, 2008.

[7] D. Needell and J. A. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *preprint, http://arxiv.org/abs/0803.2392v2*, 2008.

[8] D. Donoho, "Neighborly polytopes and sparse solution of underdetermined linear equations," *IEEE. Trans. Info. Theory*, 2006.

[9] M. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the ACM*, vol. 24, pp. 381–395, 1981.

[10] D. Donoho and J. Tanner, "Counting faces of randomly projected polytopes when the projection radically lowers dimension," *J. Amer. Math. Soc.*, posted in 2008 (to appear in print).