Robust Subspace System Identification via Weighted Nuclear Norm Optimization ⋆

Dorsa Sadigh [*] **Henrik Ohlsson** [*,**] **S. Shankar Sastry** [*]
Sanjit A. Seshia [*]

[*] *University of California, Berkeley, Berkeley, CA 94720 USA.*
{dsadigh,ohlsson,sastry,sseshia}@eecs.berkeley.edu
[**] *Division of Automatic Control, Department of Electrical Engineering, Linköping University, Sweden.*

Abstract: Subspace identification is a classical and very well studied problem in system identification. The problem was recently posed as a convex optimization problem via the nuclear norm relaxation. Inspired by robust PCA, we extend this framework to handle outliers. The proposed framework takes the form of a convex optimization problem with an objective that trades off fit, rank and sparsity. As in robust PCA, it can be problematic to find a suitable regularization parameter. We show how the space in which a suitable parameter should be sought can be limited to a bounded open set of the two-dimensional parameter space. In practice, this is very useful since it restricts the parameter space that is needed to be surveyed.

Keywords: subspace identification, robust estimation, outliers, nuclear norm, sparsity, robust PCA.

1. INTRODUCTION

Subspace system identification is a well studied problem within the field of system identification (De Moor et al., 1988; Moonen et al., 1989; Verhaegen and Dewilde, 1992a,b; Verhaegen, 1993, 1994; Van Overschee and De Moor, 1994). The problem has a nice geometrical interpretation and can be posed as a rank minimization problem. However, minimizing the rank of a matrix is NP-Hard. In recent years, there has been an increasing interest in applying the nuclear norm as a relaxation of the rank (Fazel et al., 2013; Liu and Vandenberghe, 2010; Liu et al., 2013; Hansson et al., 2012). The nuclear norm, which is the sum of singular values, gives a convex approximation for the rank of a matrix. Thus, it provides a convenient framework for system identification as well as preserving the linear structure of the matrix. In this approach, the nuclear norm of a Hankel matrix representing the data and a regularized least square fitting error is minimized. Fazel et al. (2013) study the problem of rank minimization and compare the time complexity of different algorithms minimizing the dual or primal formulation.

The problem of subspace identification with partially missing data is addressed by Liu et al. (2013), where they extend a subspace system identification problem to the scenario where there are missing inputs and outputs in the training data. The authors approach this case by solving a regularized nuclear norm optimization, where the least square fitting error is only minimized over the observed data.

Low rank problems have also been studied extensively in the areas of machine learning and statistics. One of the most studied problems is that of principal component analysis (PCA, Hotelling (1933)). Although both of the approaches, the subspace identification framework and PCA, seek low-rank structures, the major difference is the additional structure imposed in the subspace identification framework due to the linear dynamics of the system.

In this work, we extend the nuclear norm minimization framework to the case, where the output data has outliers. Our framework considers a situation where the observed sensors are attacked by a malicious agent. Thus, we would like our subspace system identification approach to be resilient to such attacks. In our solution, we formalize three tasks: (i) detecting the attack vector, (ii) minimizing the least square fitting error between our estimation and the training data, (iii) rank minimization. The attack vector is assumed to be sparse with nonzero entries corresponding to the instant of attack. We do not impose any structure on the time of attack, which is the position of outliers in the attack vector. We then estimate the attack vector as well as the model orders and model matrices. In order to impose the trade off between sparsity of the attack vector and simplicity of the structure of the system, both the attack term and the nuclear norm are penalized.

⋆ D. Sadigh is supported in part by NDSEG Fellowship, NSF grant CCF-1116993 and DOD ONR Office of Naval Research N00014-13-1-0341. This work was supported in part by TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA. H. Ohlsson gratefully acknowledge support from the NSF project FORCES (Foundations Of Resilient CybEr-physical Systems), the Swedish Research Council in the Linnaeus center CADICS, the European Research Council under the advanced grant LEARN, contract 267381, a postdoctoral grant from the Sweden-America Foundation, donated by ASEA's Fellowship Fund, and by a postdoctoral grant from the Swedish Research Council.

Our approach is inspired by the developed techniques in machine learning and robust PCA (Candès et al., 2011; Tiwari et al., 2014). The problem of robust PCA, that is to find the principal components when there exists corrupted training data points or outliers is of interest in applications like image reconstruction. The common solutions of this problem include using a robust estimator for covariance matrix.

The main contributions of the paper are twofold. The first contribution is a novel framework for robust subspace identification. The method is based on convex optimization and accurately detects outliers. The second contribution is the characterization of the regularization parameter space that needs to be surveyed. More precisely, we show that the optimization variables are zero outside a bounded open set of the two-dimensional parameter space and that the search for suitable regularization parameters can therefore be limited to this set. The derivations also apply after minor modifications to limit the search space for algorithms for robust PCA (Candès et al., 2011) and subspace identification (Fazel et al., 2013; Liu and Vandenberghe, 2010; Liu et al., 2013; Hansson et al., 2012).

In the rest of this paper, we first propose our problem setting in Section 2. We then discuss our method for detecting outliers in Section 3, and propose a heuristic for computing the penalty terms that we introduce in Section 4. In Section 5 we implement our algorithm and show the results for a dataset. We then conclude in Section 6.

2. PROBLEM FORMULATION

The problem of subspace identification can be formulated for a linear discrete-time state space model with process and measurement noise. We use the following Kalman normal form for this formulation.

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + Ke(k) \\ y(k) &= Cx(k) + Du(k) + e(k) \end{aligned} \tag{1}$$

In equation (1), we let $x(k) \in \mathbf{R}^{n_x}, u(k) \in \mathbf{R}^{n_m}, y(k) \in \mathbf{R}^{n_p}$ and $e(k) \in \mathbf{R}^{n_p}$, where $u(k)$ is the set of inputs, and $y(k)$ is the set of outputs. We let $e(k)$ be ergodic, zero-mean, white noise. Matrices A, B, C, D, K are real valued system matrices of this state-space model. The problem of subspace identification is to estimate system matrices and model order n_x, given a set of input and output traces $(u(k), y(k))$ for $k = 0, \cdots, N$.

In this work, we consider a variant of subspace identification problem, where we experience missing data and outliers in the set of output traces. Our goal is to estimate the system matrices and model orders correctly in the presence of such outliers and missing data.

Throughout this paper, we use block Hankel matrix formulation as in (Hansson et al., 2012; Liu et al., 2013) to represent equation (1).

$$Y_{0,r,N} = O_r\, X_{0,1,N} + S_r\, U_{0,r,N} + E \tag{2}$$

Here, $X_{0,1,N}$, $Y_{0,r,N}$ and $U_{0,r,N}$ are block Hankel matrices for the state, output and input sequences. A block Hankel matrix $H_{i,j,k}$ for a sequence of vectors $h(t)$ is defined to be:

$$H_{i,j,k} = \begin{bmatrix} h(i) & h(i+1) & \cdots & h(i+k-1) \\ h(i+1) & h(i+2) & \cdots & h(i+k) \\ \vdots & \vdots & \ddots & \vdots \\ h(i+j-1) & h(i+j) & \cdots & h(i+j+k-2) \end{bmatrix}$$

In equation (2), E is the noise sequence contribution, and O_r is the extended observability matrix. O_r and S_r are defined as the following matrices:

$$O_r = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{r-1} \end{bmatrix}, \quad S_r = \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \cdots & 0 \\ CAB & CB & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{r-2}B & CA^{r-3}B & \cdots & D \end{bmatrix}$$

The approach introduced by Liu et al. (2013), estimates the range space of O_r, which then can be used to attain the system matrices. First, the second term in equation (2) is eliminated by multiplying both sides of the equation by $\Pi_{0,r,N}$, an orthogonal projection matrix onto the nullspace of $U_{0,r,N}$.

$$Y_{0,r,N}\, \Pi_{0,r,N} = O_r\, X_{0,1,N}\, \Pi_{0,r,N} + E\, \Pi_{0,r,N} \tag{3}$$

As a result, in the absence of noise the following equality holds:

$$\text{range}(Y_{0,r,N}\, \Pi_{0,r,N}) = \text{range}(O_r) \tag{4}$$

If $X_{0,1,N}\, \Pi_{0,r,N}$ has full rank (which is generally the case for random inputs), it can be shown that $\text{rank}(Y_{0,r,N}\, \Pi_{0,r,N})$ is equal to n_x. Therefore, $\text{range}(O_r)$ and consequently the model order n_x can be determined by low-rank approximation of $Y_{0,r,N}\, \Pi_{0,r,N}$.

In order to guarantee the convergence of $\text{range}(Y_{0,r,N}\, \Pi_{0,r,N})$ to $\text{range}(O_r)$ as the number of input, output sequences approach infinity, a matrix Φ consisting of instrumental variables is introduced.

$$\Phi = \begin{bmatrix} U_{-s,s,N} \\ Y_{-s,s,N} \end{bmatrix} \tag{5}$$

We choose s and r to be smaller than N. To further improve the accuracy of this method, we include weight matrices W_1 and W_2. Therefore, the problem of low-rank approximation of $Y_{0,r,N}\, \Pi_{0,r,N}$ is reformulated as low-rank approximation of G.

$$G = W_1\, Y_{0,r,N}\, \Pi_{0,r,N}\, \Phi^\top\, W_2 \tag{6}$$

The weight matrices that are selected in our experiments are $W_1 = I$, and $W_2 = (\Phi\, \Pi_{0,r,N}\, \Phi^\top)^{-1/2}$ as used in PO-MOESP algorithm by Verhaegen (1994).

We approximate $\text{range}(O_r)$ to be $\text{range}(W_1^{-1}\, P)$, where P is extracted from truncating the SVD of G:

$$G = [P \ P_e] \begin{bmatrix} \Sigma & 0 \\ 0 & \Sigma_e \end{bmatrix} [Q \ Q_e]^\top \tag{7}$$

After estimation of range(O_r), we find the matrix realization of the system, and completely recover A, B, C, D and x_0.

We let $V \in \mathbf{R}^{rn_p \times n_x}$ be a matrix whose columns are a basis for the estimate of range(O_r). Then, we partition V into r block rows $V_0, \cdots, V_{r-1}$. Each of these blocks has size of $n_p \times n_x$. Therefore, the estimates of A and C are:

$$\hat{C} = V_0, \qquad \hat{A} = \arg\min \sum_{i=1}^{r-1} \|V_i \ - \ V_{i-1}\hat{A}\|_F^2 \tag{8}$$

In this equation $\| \cdot \|_F$ is the Frobenius norm. Based on the estimates $\hat{A}, \hat{C}$, it is easy to solve the following optimization problem that finds $\hat{B}, \hat{D}$ and $\hat{x}_0$.

$$\begin{aligned}(\hat{B}, \hat{D}, \hat{x}_0) = \arg\min \sum_{k=0}^{N+r-2} \|\hat{C}\hat{A}^k\hat{x}_0 + \\ \sum_{i=0}^{k-1} \hat{C}\hat{A}^{k-i}\hat{B}u(i) + \hat{D}u(k) - y(k)\|_2^2\end{aligned} \tag{9}$$

3. METHOD

Liu and Vandenberghe (2010) approach the subspace identification problem with missing data using a nuclear norm optimization technique. A nuclear norm of a matrix $||X||_* = \sum_i \sigma_i(X)$ is the sum of all singular values of matrix X, and it is the largest convex lower bound for rank(X) as shown by Fazel et al. (2001).

Therefore, given a sequence of input and output measurements, Liu et al. (2013) formulate the following regularized nuclear norm problem to estimate $\mathbf{y} = y(0), \cdots, y(N+r-2)$, which is a vector of model outputs.

$$\min_{\mathbf{y}} \|G(\mathbf{y})\|_* + \lambda \sum_{k \in T_o} \|y(k) - y_{meas}(k)\|_2^2 \tag{10}$$

In equation (10), T_o is the set of time instances for observed output sequences and $T_o \subseteq T$, where $T = \{0, \cdots, N+r-2\}$. Here, $y_{meas}(k)$ are the set of measured outputs and $k \in T_o$. The first element of the objective function is the nuclear norm of $G(\mathbf{y}) = W_1 \, Y_{0,r,N} \, \Pi_{0,r,N} \, \Phi^\top \, W_2$, as it is derived in equation (6).

Our approach, in detecting output outliers in the training data is built based on the introduced technique. We assume that the vector $\mathbf{y_{meas}}$, the measured output vector, has a sparse number of outliers or attacked output values. We do not make any extra assumptions on the specific time that the outliers will occur. Thus, we can extend equation (10) by introducing an error term $v(k) \in \mathbf{R}^{n_p}$ for $k \in T$. This error term is intended to represent the outlier present at time k; therefore, we would like vector $\mathbf{v}$ to be sparse and its non-zero elements detect the time and value of the outliers. Then, the objective we would like to minimize is:

$$\begin{aligned}&\min_{\mathbf{y},\mathbf{v}} \ f(\mathbf{y}, \mathbf{v}; \lambda_1, \lambda_2) = \\ &\min_{\mathbf{y},\mathbf{v}} \ \lambda_1 \|G(\mathbf{y})\|_* + \sum_{k \in T} \|y(k) - y_{meas}(k) - v(k)\|_2^2 \\ &\qquad + \lambda_2 \sum_{k \in T} \|v(k)\|_1\end{aligned} \tag{11}$$

In this formulation, we would like to estimate $\mathbf{y}$ and find the error term $\mathbf{v}$, such that the error vector is kept sparse, and accounts for the outliers that occur in training data $\mathbf{y_{meas}}$. The first term in this formulation is the nuclear norm with a penalty term λ_1 enforcing the low rank representation. The second term is the least square error as before, which enforces $\mathbf{v}$ to capture the outlier values in $\mathbf{y_{meas}}$. The last term is the ℓ_1-norm enforcing the sparsity criterion on vector $\mathbf{v}$. We penalize the ℓ_1-norm with λ_2.

Using the formulation in equation (11), allows us to:

(1) find a filtered version of the output measurements. In this filtered version of the output, the effects of outliers have been removed and the values for the missing data are filled in.
(2) In addition, (11) allows us to get an estimate of the value and time the outliers are appeared in the measurements. This is a valuable piece of information if the time of attack is a variable of interest.

Having recovered the filtered output, any subspace identification method could be applied to estimate the model matrices A, B, C, D and K.

4. PENALTY COMPUTATION

In this section, we discuss how to choose the penalty terms λ_1 and λ_2 in equation (11). Notice that a large enough λ_1 will force $\mathbf{y} = 0$ and a large enough λ_2 drives $\mathbf{v} = 0$.

In fact, it can be shown that there exist λ_1^{max} and λ_2^{max} such that whenever $\lambda_1 \geq \lambda_1^{max}$, $\mathbf{y} = 0$ and whenever $\lambda_2 \geq \lambda_2^{max}$, $\mathbf{v} = 0$. In practice, λ_1^{max} and λ_2^{max} are very useful since they give a range for which it is interesting to seek good penalty values. Having limited the search for good penalty values to an open set in the (λ_1, λ_2)-space, classical model selection techniques such as cross validation or the Akaike criterion (AIC) (Akaike, 1973) could be adapted to find suitable penalty values.

4.1 Computation of λ_1^{max} and λ_2^{max}

The optimal solution of equation (11) occur only when zero is included in the subdifferential of the objective in equation (11).

$$0 \in \partial f(\mathbf{y}, \mathbf{v}; \lambda_1, \lambda_2) \tag{12}$$

We find the values λ_1^{max} and λ_2^{max}, by solving $0 \in \partial f(\mathbf{y}, \mathbf{v}; \lambda_1, \lambda_2)$ subject to the constraints $\mathbf{y} = 0$ and $\mathbf{v} = 0$.

For simplicity, assume that $n_p = 1$. Therefore, equation (11) can be simplified:

$$\min_{\mathbf{y},\mathbf{v}} \ \lambda_2 \|G(\mathbf{y})\|_* + \sum_{k \in T} (y(k) - y_m(k) - v(k))^2 + \lambda_1 |v(k)| \tag{13}$$

This equation can be reformulated using the Huber norm (Huber, 1973):

$$\min_{\mathbf{y}} \ \lambda_2 \|G(\mathbf{y})\|_* + \sum_{k\in T} \|y(k) - y_m(k)\|_H, \tag{14}$$

where the Huber norm $\|\cdot\|_H$ is defined:

$$\|x\|_H = \begin{cases} x^2 & \text{if } |x| \le \lambda_1/2, \\ \lambda_1|x| - \lambda_1^2/4 & \text{otherwise} \end{cases} \tag{15}$$

Note that $\|\cdot\|_H$ is differentiable. Now, to find λ_1^{max} and λ_2^{max} we seek the smallest λ_2 such that 0 belongs to the subdifferential of the objective function with respect to $\mathbf{y}$ evaluated at $\mathbf{y} = 0$.

$$0 \in \partial_{\mathbf{y}} \Big(\lambda_2 \|G(\mathbf{y})\|_* + \sum_{k\in T} \|y(k) - y_m(k)\|_H \Big)\Big|_{\mathbf{y}=0} \tag{16}$$

This subdifferential with respect to $y(t)$ can be calculated:

$$\partial_{\mathbf{y}(t)} \Big(\lambda_2 \|G(\mathbf{y})\|_* + \sum_{k\in T} \|y(k) - y_m(k)\|_H \Big) \tag{17a}$$

$$= \lambda_2 \partial_{\mathbf{y}(t)} \Big(\|G(\mathbf{y})\|_* \Big) + \partial_{\mathbf{y}(t)} \Big(\|y(t) - y_m(t)\|_H \Big) \tag{17b}$$

In equation (6), we defined $G(\mathbf{y})$. Since, we chose W_1 to be the identity matrix I, it is reasonable to assume that $G(\mathbf{y})$ takes the form:

$$G(\mathbf{y}) = Y_{0,r,N}\, \mathbf{B} \tag{18}$$

Thus, the subdifferential of the nuclear norm in equation (17) evaluated at $\mathbf{y} = 0$ is:

$$\begin{aligned} \partial_{\mathbf{y}(t)} \Big(\|G(\mathbf{y})\|_* \Big)\Big|_{\mathbf{y}=0} &= \lambda_2 \sum_{i,j} \mathbf{V}(i,j) \partial_{\mathbf{y}(t)} \Big(G(\mathbf{y})(i,j) \Big)\Big|_{\mathbf{y}=0} \\ &= \sum_{k=1}^{t} \mathbf{V}(t-k+1,:) \mathbf{B}(k,:)^{\mathsf{T}} \\ &\quad \text{where } \|\mathbf{V}\| \le 1 \end{aligned} \tag{19}$$

See Watson (1992) and Recht et al. (2010) for the calculation of the subdifferential of the nuclear norm.

Equation (19) is analyzed for $t = 1, \ldots, N + r - 1$. We calculate this subdifferential separately for three different intervals of t: (i) $t = 1, \ldots, r$, (ii) $t = r + 1, \ldots, N$, (iii) $t = N + 1, \ldots, N + r - 1$ due to the structure of the block Hankel matrix $Y_{0,r,N}$.

Furthermore, we calculate the subdifferential of the second part of equation (17):

$$\partial_{\mathbf{y}(t)} \Big(\|y(t) - y_m(t)\|_H \Big) = \begin{cases} 2\big(y(t) - y_m(t)\big) & \text{if } |y(t) - y_m(t)| \le \lambda_1/2, \\ \lambda_1 \text{sgn}\big(y(t) - y_m(t)\big) & \text{otherwise} \end{cases} \tag{20}$$

Combining the two parts in equations (19) and (20), we rewrite the subdifferential of the objective function.

$$0 \in \lambda_2 \sum_{k=1}^{t} \mathbf{V}(t-k+1,:) \mathbf{B}(k,:)^{\mathsf{T}} - \begin{cases} 2y_m(t) & \text{if } |y_m(t)| \le \lambda_1/2 \\ \lambda_1 \text{sgn}\big(y_m(t)\big) & \text{otherwise} \end{cases}, \quad \|\mathbf{V}\| \le 1 \tag{21}$$

We hence find λ_2^{max} (for each value of λ_1) by solving the following convex program:

$$\begin{aligned} \lambda_2^{max} &= \arg\min_{\mathbf{y}} \|\mathbf{V}\| \\ \text{subj. to} \quad 0 &= \sum_{k} \mathbf{V}(t-k+1,:) \mathbf{B}(k,:)^{\mathsf{T}} - \begin{cases} 2y_m(t) & \text{if } |y_m(t)| \le \lambda_1/2 \\ \lambda_1 \text{sgn}\big(y_m(t)\big) & \text{otherwise} \end{cases} \\ &\quad \text{for } t = 1, \ldots, r. \end{aligned} \tag{22}$$

Remark 1. Note that if λ_1 is chosen such that $|y_m(t)| \le \lambda_1/2$, for $t = 1, \ldots, N + r - 1$, then $\mathbf{v} = 0$ solves (11). Therefore, $\lambda_1^{max} = 2 \max_t |y_m(t)|$.

Remark 2. In all our calculations, λ_2^{max} is a function of λ_1. From now on, we refer to λ_2^{max} as the value of this function evaluated at $\lambda_1 = \lambda_1^{max}$.

Therefore, based on remarks (1) and (2) and by solving equation (22) we find λ_1^{max} and λ_2^{max} for a given sequence of inputs and outputs

4.2 Finding the knee of the residual curve

For a given λ_1 and λ_2, the residual training error is the sum of squared residual errors at every time step of the training data, which is the difference between the measured $y_m(t)$ and the simulated $\tilde{y}(t)$ and the error term $v(t)$. The simulated $\tilde{y}(t)$ is the output of simulation of the dynamics after estimating the system matrices based on equations (8) and (9). The term $v(t)$ encodes the position and amount of outliers that occur in the training measured data $y_m(t)$.

$$\text{Residual Training Error} = \sum_{t\in T} (\tilde{y}(t) + v(t) - y_m(t))^2 \tag{23}$$

Since we have found λ_1^{max} and λ_2^{max}, we are now able to grid over the intervals $(0, \lambda_1^{max}]$ and $(0, \lambda_2^{max}]$, and calculate the residual training error for every point in the grid. Figure 1 represents this residual error for combinations of $(\lambda_1, \lambda_2) \in (0, \lambda_1^{max}] \times (0, \lambda_2^{max}]$. We linearly grid each one of the intervals the two penalty terms lie in. In this example given $\lambda_1^{max} = 29.6682$ and $\lambda_2^{max} = 766.8142$, we pick 20 linearly spaced values for each λ_1 and λ_2 as shown in Figure 1.

Given a fixed λ_1, the graph of the residual training error increases as λ_2 increases. We would like to pick a value for λ_2 that minimizes this error; however, we must avoid overfitting which can be caused by picking the smallest possible λ_2. Therefore, we select λ_2 such that it is at the knee of the curve. The knee of a curve is the point on the curve where the rate of performance gain starts diminishing, which is the area with the maximum curvature. We select the most effective λ_2 (or λ_1) by

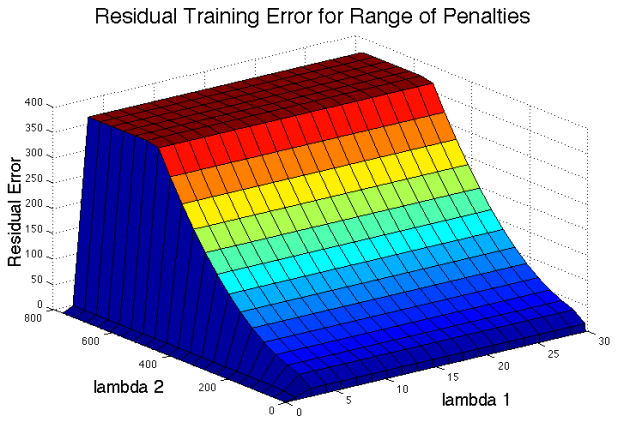

Fig. 1. Plot of the residual training error for combination of (λ_1, λ_2). The residual training error increases as both λ_1 and λ_2 are increasing.

choosing the knee of the residual training error curve for a fixed λ_1 (or λ_2).

4.3 Cross Validation

Another approach that could be used to choose λ_1 and λ_2 is to perform the calculations in Section 4.1 only on, say, 90% of the data, and calculate the residual error on the validation data which is only the 10% of the dataset that is not used in training. We assume there are no outliers in the validation data; however, subspace system identification cannot be done solely on this batch of data since the 10% validation batch does not have sufficient number of data points for a complete subspace system identification. After the computation of λ_1^{max} and λ_2^{max}, we choose a grid for combinations of (λ_1, λ_2) as before. We then calculate the residual validation error for every pair of (λ_1, λ_2).

$$\text{Residual Validation Error} = \sum_{t \in T_{valid}} (\tilde{y}(t) - y_{valid}(t))^2 \tag{24}$$

The residual validation error is the sum of squared error, where T_{valid} represents the data points for the 10% validation batch, and $y_{valid}(t)$ is the measured output for this batch. As in equation (23), $\tilde{y}(t)$ is the output of simulation of dynamics for the specific time window T_{valid}.

In this case, we choose the set of (λ_1, λ_2) that minimize the residual validation error. As both penalty terms get smaller the validation error drops as well; however, this error reaches a minimum and starts increasing as the penalty terms get smaller due to over fitting. Therefore, the most effective set of (λ_1, λ_2) are the largest pair that drive the validation error to its minimum.

5. EXPERIMENTAL RESULTS

In our experiments we use the data from the DaISy database by De Moor et al. (1997) and insert outliers at randomly generated indices in the output set. We then perform our algorithm to detect the indices with outliers and recover the output $\hat{\mathbf{y}}$ as well as the model matrices. Table 5 and Figure 2 correspond to data of a simulation of an ethane-ethylene destillation. We let $r = 5$, and $s = 5$, and take the first 5 input and output values of this benchmark as instrumental variables. We use the rest of this data sequence as $\mathbf{y_{meas}}(t)$, where $t \in \{1, \ldots, 85\}$ and $T = 85$. Then outliers are inserted at randomly chosen indices of $\mathbf{y_{meas}}$. The insertion of outliers is either by subtracting or adding a large value to a randomly selected index of vector $\mathbf{y_{meas}}(k)$. For the destillation benchmark, $n_p = 3$, that is $\mathbf{y_{meas}}(k) \in \mathbb{R}^3$. Thus, for a randomly selected time index $k \in \{1, \ldots, 85\}$, we randomly choose one of the vector elements of $\mathbf{y_{meas}}(k)$, and either add or subtract a large value (in our example 20 since the elements of $\mathbf{y_{meas}}$ range from $[-9.5267, 7.2139]$). Based on the calculations for computation of penalties in Section 4, we select $\lambda_1 = 1$ and $\lambda_2 = 1$.

We define rate of correct detection as the ratio of correctly detected outliers to the number of true outliers. The correctly detected outliers are the number of outliers detected at the same exact indices as the true outliers. We first set the number of true outliers to 3 in a dataset with 85 points, and perform a Monte Carlo simulation with 50 iterations. In every iteration, we insert three randomly selected outliers in the dataset. We report the mean value of rate of correct detection and the computation time of one iteration in Table 5 for the desillation benchmark. As the numbers suggest, the rate of correct detection decreases as the noise level of this dataset increases.

Noise Level	Rate of Outlier Detection	Time(s)
No noise	0.9800	18.6269
10% noise	0.9467	18.0497
20% noise	0.8933	18.5571
30% noise	0.9000	15.1469

Table 1. Rate of correct outlier detection for ethane-ehylene destillation benchmark with different levels of noise. The penalties are chosen to be $\lambda_1 = 1$ and $\lambda_2 = 1$.

We then calculate the rate of correct detection for different number of outliers in the dataset. Figure 2 shows the drop in rate of correct detection of outliers as the number of outliers increase in the dataset. Similar to before, we perform 50 iterations of Monte Carlo Simulation and plot the mean value of rate of detection. In every iteration, a new set of randomly generated indices were selected for inserting outliers. We range the number of outlier insertions from 3 to 50 points for a dataset with $T = 85$ data points.

We did not encounter any false positives, where the algorithm detects an incorrect index as an outlier point, in any of these iterations. The inexact rate of detection in Figure 2 corresponds to failure in detection of an outlier in every iteration rather than misdetection.

With our algorithm, we are able to correctly detect outliers. We then use any proposed subspace system identification method to find the model matrices. We follow the same approach proposed by Liu et al. (2013), that is to use the estimation of the output and create the estimated output Hankel matrix $\hat{Y}_{0,r,N}$. Then range(O_r) the extended observability matrix is evaluated by applying SVD on the estimated $\hat{G}$ as in equation (6).

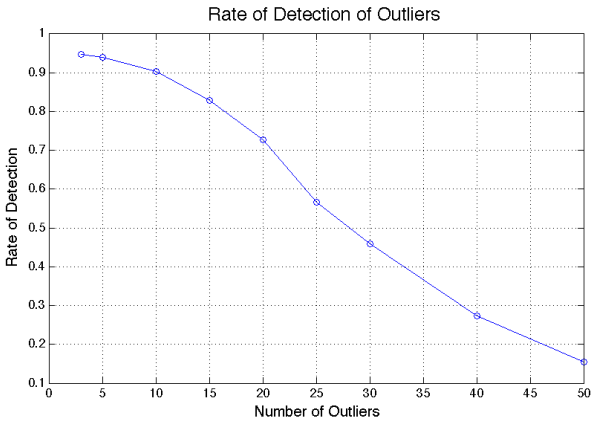

Fig. 2. This plot shows the rate of detection of outliers for a given number of inserted outliers. The rate of detection decreases as we insert more outliers in the data.

$$\hat{G} = W_1 \, \hat{Y}_{0,r,N} \Pi_{0,r,N} \, \Phi^\top \, W_2 \tag{25}$$

We apply the optimizations in equations (8) and (9) to find the estimated model matrices and the estimated initial state. Finally, we have completely realized the system matrices from a dataset with randomly attacked output values.

6. CONCLUSION AND FUTURE WORK

In this paper, an outlier-robust approach to subspace identification was proposed. The method takes the form of a convex optimization problem and was shown to accurately detect outliers. The method has two tuning parameters trading off the sparsity of the estimate for outliers, the rank of a system matrix (essentially the order of the system) and the fit to the training data. To aid in the tuning of these parameters, we show the search for suitable parameters can be restricted to a bounded open set in the two dimensional parameter space. This can be very handy in practice since an exhaustive search of the two-dimensional parameter space is often time consuming. we note that this way of bounding the parameter space could also be applied to robust PCA. This is left as future work. To further speed up the framework, the alternating direction method of multipliers (ADMM) could be used to solve the optimization problem. This was not considered here but is seen as a possible direction for future work.

REFERENCES

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In *Proceedings of the 2nd International Symposium on Information Theory*, 267–281. Akademiai Kiado, Budapest.

Candès, E.J., Li, X., Ma, Y., and Wright, J. (2011). Robust principal component analysis? *J. ACM*, 58(3), 11:1–11:37.

De Moor, B., Moonen, M., Vandenberghe, L., and Vandewalle, J. (1988). A geometrical approach for the identification of state space models with singular value decomposition. In *Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on*, 2244–2247 vol.4.

De Moor, B., De Gersem, P., De Schutter, B., and Favoreel, W. (1997). Daisy: A database for identification of systems. *Journal A*, 38, 4–5.

Fazel, M., Pong, T., Sun, D., and Tseng, P. (2013). Hankel matrix rank minimization with applications to system identification and realization. *SIAM Journal on Matrix Analysis and Applications*, 34(3), 946–977.

Fazel, M., Hindi, H., and Boyd, S.P. (2001). A rank minimization heuristic with application to minimum order system approximation. In *Proc. of the American Control Conference*.

Hansson, A., Liu, Z., and Vandenberghe, L. (2012). Subspace system identification via weighted nuclear norm optimization. *CoRR*, abs/1207.0023.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(7), 498–520.

Huber, P.J. (1973). Robust regression: Asymptotics, conjectures and Monte Carlo. *The Annals of Statistics*, 1(5), 799–821.

Liu, Z., Hansson, A., and Vandenberghe, L. (2013). Nuclear norm system identification with missing inputs and outputs. *Systems & Control Letters*, 62(8), 605–612.

Liu, Z. and Vandenberghe, L. (2010). Interior-point method for nuclear norm approximation with application to system identification. *SIAM Journal on Matrix Analysis and Applications*, 31(3), 1235–1256.

Moonen, M., De Moor, B., Vandenberghe, L., and Vandewalle, J. (1989). On- and off-line identification of linear state space models. *International Journal of Control*, 49, 219–232.

Recht, B., Fazel, M., and Parrilo, P. (2010). Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3), 471–501.

Tiwari, A., Dutertre, B., Jovanovic, D., de Candia, T., Lincoln, P., Rushby, J., Sadigh, D., and Seshia, S. (2014). Safety envelope for security. In *Proceedings of the 3rd International Conference on High Confidence Networked Systems*, HiCoNS '14. ACM.

Van Overschee, P. and De Moor, B. (1994). N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1), 75 – 93.

Verhaegen, M. (1993). Subspace model identification part 3. analysis of the ordinary output-error state-space model identification algorithm. *International Journal of control*, 58(3), 555–586.

Verhaegen, M. and Dewilde, P. (1992a). Subspace model identification part 1. the output-error state-space model identification class of algorithms. *International journal of control*, 56(5), 1187–1210.

Verhaegen, M. and Dewilde, P. (1992b). Subspace model identification part 2. analysis of the elementary output-error state-space model identification algorithm. *International journal of control*, 56(5), 1211–1241.

Verhaegen, M. (1994). Identification of the deterministic part of MIMO state space models given in innovations form from input-output data. *Automatica*, 30(1), 61 – 74.

Watson, G. (1992). Characterization of the subdifferential of some matrix norms. *Linear Algebra and its Applications*, 170(0), 33–45.