Set-Valued Estimators for Uncertain Linear Parameter-Varying Systems

Daniel Silvestre[a,b,c]

[a]*School of Science and Technology from the NOVA University of Lisbon (FCT/UNL), 2829-516 Caparica, Portugal.*
[b]*COPELABS, Lusófona University, Lisbon, Portugal.*
[c]*Institute for Systems and Robotics, Instituto Superior Técnico, University of Lisbon, Lisbon, Portugal.*

Abstract

In this paper, we tackle the problem of state estimation for uncertain linear systems when bounds are known for the disturbances, noise and initial state. Practical systems often have parameters that cannot be measured precisely at every iteration. The framework of Uncertain Linear Parameter-Varying systems (Uncertain LPVs) have attracted attention from the community and have seen applications from the aerospace industry to mechatronic systems, among many other examples. By formulating the problem as the solution of a feasibility program, we show that the optimal convex solution can be computed through an enumeration of the vertices of the estimates. Resorting to this result, three algorithms are proposed: an approximation algorithm using only set operations; an exact convex hull method returning the optimal convex set suitable for cases where estimates do not have a large number of vertices; and an event-triggering algorithm suitable for fault/attack detection that combines both the convex and nonconvex methods. Simulations are conducted using a motor speed model where some of the parameters cannot be measured exactly pointing out that the uncertainty matrices are responsible for the accuracy of the approximation algorithm, and also that the point-based method is suitable for online estimation.

Keywords: State estimation; set-membership approaches; uncertain LPVs; fault detection.

1. Introduction

In this paper, the problem of estimating the state of a dynamical system for a broad family of linear systems is tackled. The task has been addressed by two main approaches: i) stochastic — where some information regarding the probability distribution is assumed to be known, with examples such as the well-established Kalman Filter and its variants; ii) set-membership — where bounds are known for the values of the unknown signals with a large body of research considering different types of bounds and representation descriptions for the sets.

Linear Parameter-Varying (LPV) models have been introduced by the work of Michael Athans (for example see [1]) to encompass a class of nonlinear dynamics that can be treated as linear systems when designing controllers and observers. These models have a variety of applications in aerospace industry, mechatronic systems, automotive, robotic manipulators, vehicle motion, active magnetic bearings, among other academic examples as reported in the survey [2]. We remark that parameters in LPV are not known only at the design phase. However, when parameters cannot be measured during execution, the family is called Uncertain LPVs. These types of systems are radically different from standard Linear Time-Varying (LTV), where the entries are known functions over time. The major advantage is that one can treat a subset of nonlinear dynamics whenever these nonlinear parameters can be measured (LPVs) or account for model inaccuracies and approximation residuals (Uncertain LPVs).

The estimation task for LTVs is well established using interval arithmetic [3, 4], zonotopes [5], ellipsoids [6], constrained zonotopes (following a trivial extension from the work in [7]), polytopes [8] and even by combining different Convex Generators [9]. On the other hand, for nonlinear systems these strategies can be extended through the use of approximation functions to the nonlinear dynamics and using the same types of set description as for the LTVs as in [10], [11], [12], [13], [14], respectively. However, by explicitly considering an Uncertain LPVs it makes possible for tighter estimation sets than for general nonlinear dynamics, which represents a gap in the literature. The main challenge is that uncertainty parameters in the dynamics represent bilinear constraints that cannot be directly represented using any of the set representations.

In the literature, the main approach to solving the estimation problem for Uncertain LPVs uses polytopes for each of the vertices of the uncertainty polytope, followed by a convex hull computation of all the produced sets [15] and can resort to a coprime factorization to decrease the impact of the initial uncertainty whenever using the approach in a model invalidation problem (such as the case of fault detection or model selection) [16] [17]. However, this approach has an exponential complexity in the worst-case by having to first generate all vertices for the polytopic uncertainty, compute all polytopes for each of the vertices (there can be an exponential number of them), followed by a convex hull of all the sets. In this paper, we first formalize the problem in order to assess some of its fundamental limitations and propose a technique based on constrained zonotopes (a similar one could be defined using the hyper-plane definition of polytopes given that they are equivalent formulations [7]) to perform the set-valued estimation. The optimal solution and its relationship to the proposed method are also discussed.

Therefore, the main contributions of this paper can be summarized as:

- The optimal solution to the set-valued estimation of Uncertain LPVs is formulated as a feasibility problem;
- We show that performing the convex hull for all polytopes or constrained

zonotopes obtained using all combinations of uncertainty vertices is the optimal convex solution to the problem;

- A novel method based on constrained zonotopes is proposed to replace the bilinear constraints as an approximated solution, which is the optimal if one is restricted to set operations;

- An efficient and exact convex hull method is proposed that has performance enabling it to be applied to online estimation of the state, i.e., such that its computation time is smaller than typical sampling times;

- Lastly, we note that for fault detection/isolation or to detect attackers in the system, an event-triggering mechanism based on the elapsed time can be employed that resorts to the nonconvex solution for the detection between triggering times and resets the constraints at triggering times using the proposed convex hull method.

The remainder of the paper is organized as follows. In Section 2, we formalize the problem as a feasibility program and point out a solution to find the convex hull of the generally nonconvex set. Three different algorithms are presented in Section 3 while pointing out their relationship with the optimal set. Simulations for a motor speed control model are presented in Section 4 and final conclusions and directions of future work as presented in Section 5.

Notation: In this paper, we denote by $\mathbf{v}$ an anonymous variable in an optimization problem that corresponds to a possible value for the vector v. The Minkowski sum of two sets X and Y is defined as $X \oplus Y := \{v + u : v \in X, u \in Y\}$. The convex hull function that outputs a hyper-plane representation of the smallest polytope enclosing all points in set A is given as $\text{convHull}(A)$. Function $\text{vertex}(X)$ returns a set of all vertices of the polytope X. The infinity norm of a vector is denoted by $\|v\|_\infty$ and corresponds to $\max_i |v_i|$ for the absolute value function $|a|$ for the scalar a. We use $\text{rank}(A)$ to denote the dimension of the column space of matrix A.

2. Problem Formulation

The problem of state estimation in Uncertain LPVs in the set-membership approach consists in finding a set of possible values given the dynamics and measurements obtained from the system. These models can be written as:

$$\begin{aligned} x(k+1) &= \Big(A(\rho(k)) + \sum_{\ell=1}^{n_\Delta} \Delta_\ell(k) U_\ell\Big) x(k) + B(\rho(k)) u(k) + L(\rho(k)) d(k) \\ y(k) &= C(\rho(k)) x(k) + N(\rho(k)) w(k) \end{aligned} \tag{1}$$

where $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^{n_u}$, $d(k) \in \mathbb{R}^{n_d}$, $y(k) \in \mathbb{R}^m$ and $w(k) \in \mathbb{R}^{n_w}$ are the system state, input, disturbance signal, output and noise, respectively. The parameter $\rho(k)$ is the part of the parameters that can be measured at time

k, which do not pose any additional difficulties for the estimation using a set-membership approach. The main challenge appears from considering the n_Δ uncertainties denoted by Δ_ℓ and the constant matrices U_ℓ that account for how the uncertainties affect the nominal dynamics matrix given by $A(\rho(k))$. To lighten the notation, we will consider $A_k := A(\rho(k))$ and similarly for all the remaining matrices in (1). Moreover, in order to have a well-posed problem, we assume that all unknown signals are bounded within a compact convex polytope denoted by the correspondent capital letter, i.e., $x(0) \in X(0)$, $d(k) \in D(k)$ and $w(k) \in W(k)$. Without loss of generality, the scalar uncertainty parameters Δ_ℓ satisfy $|\Delta_\ell| \leq 1$.

The problem addressed in this paper is summarized as:

Problem 1. *Given compact polytopic sets $X(0)$, $D(k)$ and $W(k)$ for all $k \geq 0$ and measurements $y(k)$, how to compute a set $X(k)$ such that it is guaranteed that $x(k) \in X(k)$, $\forall k \geq 0$.*

Notice that Problem 1 is called *state estimation* although a converse definition could be presented for the output of the system (this is of particular interest in sensitivity analysis [18] and system distinguishability [19]).

The first step in formalizing the problem is through the description of possible solutions. Verifying if a given point $p \in \mathbb{R}^n$ belongs to $X(k)$ is equivalent to solving the following feasibility problem:

$$\begin{array}{cll} \underset{\substack{\mathbf{x}(0)\cdots\mathbf{x}(k),\\ \mathbf{d}(0)\cdots\mathbf{d}(k-1)\\ \mathbf{w}(1)\cdots\mathbf{w}(k)\\ \boldsymbol{\Delta}_{\mathbf{1}}(0)\cdots\boldsymbol{\Delta}_{\mathbf{1}}(k-1)\\ \vdots\\ \boldsymbol{\Delta}_{\mathbf{n}_{\boldsymbol{\Delta}}}(0)\cdots\boldsymbol{\Delta}_{\mathbf{n}_{\boldsymbol{\Delta}}}(k-1)}}{\text{find}} & & \\ \text{s.t.} & \mathbf{x}(0) \in X(0), & \\ & \mathbf{d}(i) \in D(k) & , 0 \leq i \leq k-1, \\ & \mathbf{w}(i) \in W(k) & , 1 \leq i \leq k, \\ & |\boldsymbol{\Delta}_\ell(i)| \leq 1 & , 0 \leq i \leq k-1, 1 \leq \ell \leq n_\Delta, \\ & \mathbf{x}(k) = p, & \\ & \mathbf{x}(i) \text{ satisfy } (1), 0 \leq i \leq k & \end{array} \tag{2}$$

The feasibility problem in (2) is written with $\mathbf{x}$ variables accounting for the possible values of x for each of the time instants, and a similar notation for the remaining variables. The problem has a set of convex constraints and the last one is bilinear since it involves the product of $\boldsymbol{\Delta_\ell}$ and $\mathbf{x}$.

In the next theorem, we show that, if the set of all points p that satisfy (2) is a convex set, the solution can be computed using a point-based method.

Theorem 2. *Let $\Theta(k)$ be the optimal set to the estimation problem defined as $\Theta(k) = \{p : \forall p$ satisfies (2)$\}$ for any given time instant k. If $\Theta(k)$ is convex then:*

$$i)\ \Theta(k) = \text{convHull}\left(\bigcup_{\substack{v_x \in \text{vertex}(X(k-1)) \\ v_{\Delta_\ell} \in \{-1,1\} \\ v_d \in \text{vertex}(D(k-1))}} \left(A_{k-1} + \sum_{\ell=1}^{n_\Delta} v_{\Delta_\ell} U_\ell\right) v_x + B_{k-1}u(k-1) + L_{k-1}v_d \right) \bigcap Y(k),$$

where $Y(k) := \{q : y(k) = C_k q + N_k w(k), w(k) \in W(k)\}$.

Proof. We first notice that the solution to (2) can be given as:

$$\Theta(k) = X_p(k) \oplus B_{k-1}u(k-1) \oplus L_{k-1}D(k-1) \bigcap Y(k) \tag{3}$$

where $X_p(k)$ corresponds to the set of all points propagated using all possible instances of the uncertain dynamics matrices, the $\oplus$ notation stands for the Minkowski sum of sets and $Y(k)$ corresponds to the set of possible state vectors that would result in the obtained $y(k)$. By assumption, all signals are assumed to take values in compact convex sets and, therefore, the sets $B_{k-1}u(k-1)$, $L_{k-1}D(k-1)$ and $Y(k)$ are all convex since they are the result of applying a linear map to convex compact sets. If $\Theta(k)$ is convex, then $X_p(k)$ must be convex since the Minkowski sum and intersection operations preserve convexity.

If $X_p(k)$ is convex, it forms a convex polytope of matrices and one can replace the bilinear constraint by the convex hull of the sets produced by a linear constraint for each vertex of the set $X(k-1)$ [1]. Let us recall that:

$$A \oplus B = \text{convHull}\left(\bigcup_{v_a \in \text{vertex}(A), v_b \in \text{vertex}(B)} v_a + v_b \right)$$

for two polytopes A and B. Thus, using the format in (3) and the definition of $X_p(k)$ after replacing the bilinear constraints by the union of linear constraints for all vertices of $X(k-1)$, the conclusion follows. ∎

Theorem 2 draws an important fact regarding the state estimation problem for Uncertain LPVs, namely that if the optimal set is convex it will be a polytope given the assumption that the initial state, disturbance and noise signals are contained within polytopes. The following corollary is also useful.

Corollary 3. *The optimal convex solution $\Theta(k)$ to the feasibility problem in (2) is a convex polytope.*

[1] Please see the implemented Yalmip example in https://yalmip.github.io/example/lpvstatefeedback/

Corollary 3 asserts that the Set-Valued Observers (SVOs) computation is optimal for Uncertain LPVs, which extends the result in [20] for LTV systems. This is one of the main contributions of this paper in showing that a point-based method using the vertices produces the optimal convex set enveloping the solution of (2). The SVO algorithm works by computing a polytopic set for each vertex of the uncertainty polytope and doing the convex hull of the union of all such sets. However, the algorithm proposed in [8] requires twice the number of constraints than is necessary, leading to a worse efficiency. A point-based algorithm corresponding to the result in Theorem 2 to compute the optimal convex solution set is presented in pseudo-code in Algorithm 1.

Algorithm 1 State estimation for Uncertain LPVs using the vertices of the polytopes.

Require: Set $X(0)$ and, for all $k \geq 0$, sets $D(k)$, $W(k)$ and measurement polytope $Y(k)$.

Ensure: Computation at each time instant k of $X(k)$ as the convex hull of the list of points stored in the variable plist.

1: **for each** k **do**
2:     plist $= \emptyset$
3:     /* *Find the vertices of the necessary sets* */
4:     $V_x = \text{vertex}(X(k-1))$
5:     $V_d = \text{vertex}(D(k-1))$
6:     /* *For each combination of vertices find the propagated point* */
7:     **for each** $v_x \in V_x$ **do**
8:         **for each** $v_d \in V_d$ **do**
9:             **for each** $v_\Delta \in \{-1,1\}^{n_\Delta}$ **do**
10:                 plist $=$ plist $\cup \left(A_k + \sum_{\ell=1}^{n_\Delta} v_{\Delta_\ell} U_\ell\right) v_x + B_k u(k) + L_k v_d$
11:             **end for**
12:         **end for**
13:     **end for**
14:     /* *Create propagated polytope* */
15:     $X_p(k) = \text{convHull}\,(\text{plist})$
16:     /* *Update the propagated polytope* */
17:     $X(k) = X_p(k) \cap Y(k)$
18: **end for**

The main disadvantage of Algorithm 1 is that it requires enumerating all vertices of the polytopes (be it saved in the hyper-plane representation as in [8] or its constrained zonotope format [7]), which in the worst-case can represent an exponential growth followed by a combinatorial computation done in line 10 within the for cycles. However, there are very efficient algorithms to compute the convex hull in line 15, which makes the algorithm particularly suitable to cases where the sets $D(k)$ are known *a priori* and preferably constant over time. In such cases, the vertices can be computed offline and stored for future uses.

In order to illustrate to the reader the results presented in this section, we

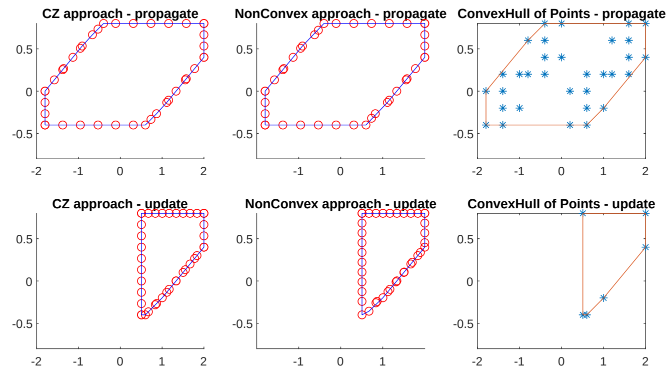

Figure 1: The produced polytopes for the example in (4) with $k = 1$ using the approximation algorithm in Section 3 both for the propagated set (up) and the updated set (bottom) are given on the left. The same sets are given using the feasibility approach in (2) (middle) and with Algorithm 1 (right).

have considered a simple model given by:

$$\begin{aligned} x(k+1) &= \left(\begin{bmatrix} 0.2 & 0.5 \\ 0.1 & 0.3 \end{bmatrix} + \Delta_1(k) \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \right) x(k) + \begin{bmatrix} 0.1 \\ 0.2 \end{bmatrix} u(k) + \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix} d(k) \\ y(k) &= \begin{bmatrix} 1 & 0 \end{bmatrix} x(k) + w(k) \end{aligned} \tag{4}$$

with $\|x(0)\|_\infty \leq 1$, and for all $k \geq 0$ the disturbance and noise signals were considered to satisfy $\|d(k)\|_\infty \leq 1$ and $\|w(k)\|_\infty \leq 1$. Also, $\Delta_1(k) \in [-1, 1]$ for all $k \geq 0$. We implemented a solution based on the constrained zonotopes description to be found in Section 3 along with the optimal feasibility set in (2) and the algorithm described in Algorithm 1. The produced sets for $X(1)$ are depicted in Figure 1 where the circles correspond to the grid points used to draw the boundary of the polytope and the asterisks on the convex hull approach corresponds to all points within plist of Algorithm 1. Given that the set $X(0)$ possesses a symmetry to be discussed in Section 3.1, the sets computed by the approximation algorithm are the optimal sets produced both by the points approach or the feasibility method.

In order to better illustrate the difference, we depict in Figure 2 the propagated and updated sets for the three algorithms at time $k = 2$ and similarly in Figure 3 for $k = 3$. Interestingly, for $k = 2$ the optimal solution of the nonconvex approach is a convex set, and we obtain the same set using Algorithm 1. However, for $k = 3$, the optimal set is no longer convex but Algorithm 1 finds its convex hull. The approximation method (in the left), is more conservative but with a lower computational cost since it only applies set operations instead of requiring converting set representations to its vertices in each time step.

Elaborating on the complexity, the feasibility problem in (2) has a number of variables equal to $k(n+n_\Delta+n_d+n_w)$, meaning that, at iteration 8, there exists 36 variables and 37 constraints. Compiling the constraints in (2) took around 10 ms in a Hewlett Packard (HP) personal computer running Windows 10, Matlab

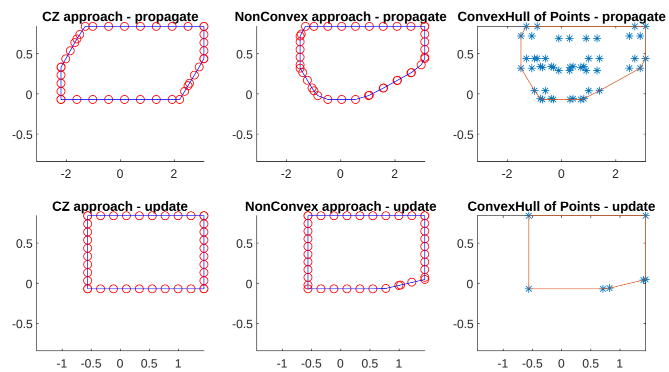

Figure 2: The produced polytopes for the example in (4) with $k = 2$ using the approximation algorithm in Section 3 both for the propagated set (up) and the updated set (bottom) are given on the left. The same sets are given using the feasibility approach in (2) (middle) and with Algorithm 1 (right).

R2018a with a processor Intel i7-8550U at 1.8GHz and with 12GB of RAM. However, checking if a point belongs to the set took on average 0.54 seconds at the $k = 1$ and at $k = 10$ was already taking 4.39 seconds using the solver BMIBNB available in Yalmip version 30-Sep-2016. Therefore, the non-convex approach is not viable unless the observer is applied in an off-line estimator. Later in this paper, we also propose the use of the non-convex approach for fault detection with a window mechanism to serve as a trade-off between accuracy and performance.

3. Set-valued Estimator based on Constrained Zonotopes

The previous section hinted at an important fact that either the state estimation task is optimal through a nonconvex approach with a fast growing complexity or the optimal convex set requires propagating individual points for all combination of vertices of all polytopes containing the unknown signals. This method is optimal in computing the convex hull for the non-convex set membership problem at the expenses of an increase in computational complexity whenever the sets have a large number of vertices or when the number of uncertainty parameters is high. In both cases, the method has a combinatorial nature of propagating points using different values corresponding to all vertices.

In the realm of LTV systems, the next set-valued estimate is equivalent to performing the propagate phase:

$$X_{prop}(k+1) = A_k X(k) \oplus B_k u(k) \oplus L_k D(k) \tag{5}$$

where a matrix multiplying a set corresponds to applying that linear map to all vectors in the set. In a similar fashion, the update step could be carried out:

$$X(k) = X_{prop}(k) \cap_{C_k} y(k) \oplus N_k W(k) \tag{6}$$

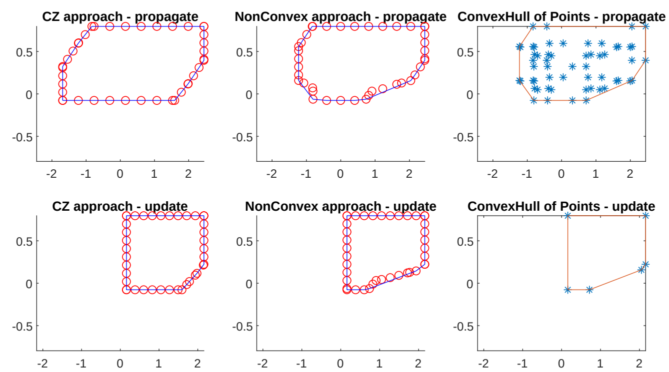

Figure 3: The produced polytopes for the example in (4) with $k = 3$ using the approximation algorithm in Section 3 both for the propagated set (up) and the updated set (bottom) are given on the left. The same sets are given using the feasibility approach in (2) (middle) and with Algorithm 1 (right).

where the symbol $\cap_{C_k}$ stands for the intersection through the map C_k such that both sets being intersected constrain the possible values of $x(k)$. We opt by representing the polytopes through a constrained zonotope formulation and, for the sake of completeness, introduce how each of the operations is defined as described in [7]. We remark to the reader that other solutions based on intervals [4] would achieve better performance by sacrificing accuracy. This is due to the fact that the sets would be overbounded by hyper-rectangles adding conservatism that would then be propagated using the dynamics for future time steps.

Definition 4 (Constrained Zonotope). *A set Z is a constrained zonotope defined by the tuple $(G, c, A, b) \in \mathbb{R}^{n \times n_g} \times \mathbb{R}^n \times \mathbb{R}^{n_c \times n_g} \times \mathbb{R}^{n_c}$ such that:*

$$Z = \{G\xi + c : \|\xi\|_\infty \leq 1, A\xi = b\}.$$

Definition 5 (Set operations). *Consider three constrained zonotopes as in Definition 4:*

- $Z = (G_z, c_z, A_z, b_z) \subset \mathbb{R}^n$;
- $W = (G_w, c_w, A_w, b_w) \subset \mathbb{R}^n$;
- $Y = (G_y, c_y, A_y, b_y) \subset \mathbb{R}^m$;

and a matrix $R \in \mathbb{R}^{m \times n}$. The three set operations are defined as:

$$RZ = (RG_z, Rc_z, A_z, b_z) \tag{7}$$

$$Z \oplus W = \left(\begin{bmatrix} G_z & G_w \end{bmatrix}, c_z + c_w, \begin{bmatrix} A_z & 0 \\ 0 & A_w \end{bmatrix}, \begin{bmatrix} b_z \\ b_w \end{bmatrix} \right) \tag{8}$$

$$Z \cap_R Y = \left(\begin{bmatrix} G_z & 0 \end{bmatrix}, c_z, \begin{bmatrix} A_z & 0 \\ 0 & A_y \\ RG_z & -G_y \end{bmatrix}, \begin{bmatrix} b_z \\ b_y \\ c_y - Rc_z \end{bmatrix} \right). \tag{9}$$

Using the set operations in Definition 5, the propagate in (5) can be implemented resorting to linear maps applied to the sets as in (7) followed by Minkowski sums of the sets as in (8). The update in (6) starts by creating the set for the measurements using both linear maps and Minkowski sums and then intersecting using (9).

We now detail three different methods, identifying the scenarios for which they are suitable. We point out to the interested reader that Constrained Zonotopes are a representation of polytopes as given in [7]. We conjecture that these sets are bounded in terms of hyper-volume following the discussion in [8] under mild stability conditions.

3.1. Approximation method

In order to deal with the uncertain component in (1), we first address the problem when $n_\Delta = 1$ and matrix U_1 satisfies $\mathrm{rank}(U_1) = 1$ such that there exist vectors e_1 and f_1 satisfying:

$$A_1 = e_1 f_1^\intercal.$$

Moreover, by defining an auxiliary vector $z_1(k) = f_1^\intercal x(k)\Delta_1(k)$ we can rewrite (1) as:

$$\begin{aligned} x(k+1) &= A_k x(k) + e_1 z_1(k) + B_k u(k) + L_k d(k) \\ y(k) &= C_k x(k) + N_k w(k) \end{aligned} \tag{10}$$

where the signal $z_1(k) \in \mathbb{R}$ is bounded by $|z_1(k)| \leq \max |f_1^\intercal x(k)|$. The right-hand side of the inequality can be approximated by solving two linear programs:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_1^\intercal \mathbf{x} \\ \text{s.t.} \quad & \mathbf{x} \in X(k) \end{aligned} \tag{11}$$

and

$$\begin{aligned} \min_{\mathbf{x}} \quad & -f_1^\intercal \mathbf{x} \\ \text{s.t.} \quad & \mathbf{x} \in X(k) \end{aligned} \tag{12}$$

and taking the maximum of both absolute values. Thus, $z_1(k) \in Z_1(k)$ which is defined as $Z_1(k) = (b, 0, 0, 0)$ where b is the maximum of the absolute value of the cost functions in programs (11) and (12). As a consequence, the system in (10) is in an LTV format with an extra unknown input $z_1(k)$ whose constrained zonotope can be computed at time k.

Remark 6. *We remark that the rank one decomposition results in the optimal set operation (i.e., without having any point-wise sum) procedure for the state estimation. By optimal, we mean that there exists at least one point for which the direction e_1 or $-e_1$ achieves the maximum bound and the other it is an outer approximation. Thus, to achieve a smaller estimation set, a point-wise operation would be required such that we could have different points in the set being affected by different values of the uncertainty set. In the example (4),* *when the previous set was symmetric with respect to the hyper-planes $f_1^\intercal \mathbf{x}$ and $e_1^\intercal \mathbf{x}$ (Figure 1) the produced set was the optimal one whereas when this failed we obtained an over-approximation (Figure 2 and Figure 3).*

The case when matrix U_1 has a rank greater than the unity means that:

$$A_1 = e_{1,1} f_{1,1}^\intercal + e_{1,2} f_{1,2}^\intercal + \cdots + e_{1,m} f_{1,r}^\intercal$$

for some $r > 0$. By defining additional variables:

$$z_{1,j}(k) := f_{1,j}^\intercal x(k) \Delta_1(k)$$

we can carry out the same procedure as for the case of a rank one matrix.

Remark 7. *We draw attention that the above separation of matrix U_1 into independent exogenous signals increases the size of the produced set as we are implicitly ignoring the relationship between the entries in A_k affected by uncertainty Δ_1.*

If $n_\Delta > 1$, the same procedure can be applied for all the remaining uncertainties as done for Δ_1 with the produced sets added by the Minkowski sum.

3.2. Exact convex hull method

The previous method explored a relaxation to the bilinear constraints imposed by the product between state and uncertainty. Such an algorithm is the optimal one using only set operations. In this section, we detail how to improve it combining both the idea in Algorithm 1 and the commutativity of the convex hull operation and the Minkowski sum. This method is of interest for cases where the set-valued estimates do not have a very large number of vertices.

Since the solution to the state estimation for Uncertain LPVs can be a nonconvex set, we opt to compute its convex hull:

$$\begin{aligned}
\text{convHull}\left(\Theta(k)\right) &= \text{convHull}\Big(X_p(k) \oplus B_{k-1}u(k-1) \oplus L_{k-1}D(k-1) \cap_{C_k} y(k) \oplus N_k W(k) \Big) \\
&= \text{convHull}\Big(X_p(k) \oplus B_{k-1}u(k-1) \oplus L_{k-1}D(k-1) \Big) \cap_{C_k} y(k) \oplus N_k W(k) \\
&= \text{convHull}\Big(X_p(k) \Big) \oplus B_{k-1}u(k-1) \oplus L_{k-1}D(k-1) \cap_{C_k} y(k) \oplus N_k W(k)
\end{aligned} \tag{13}$$

The first step in (13) used the fact that the convex hull is defined as the intersection of all convex sets enclosing the set. Since the measurement set is assumed to be convex, this intersection can be performed after the convex hull. The second step resorted to the commutativity of the convex hull and the Minkowski sum to first apply the convex hull to each set before taking the addition. Given that the actuation (a single point) and the disturbance sets are convex, its convex hull is equal to the set themselves. The formulation in (13) means that the proposed algorithm to compute the exact convex hull follows the steps:

i) Compute vertex $(X(k-1))$;

ii) Propagate all vertices from i) using the vertices -1 and 1 for each of the n_Δ uncertainties;

iii) Compute the convex hull of ii);

iv) Use the constrained zonotope set operations in Definition 5 to compute convHull $(\Theta(k))$ following (13).

In the proposed algorithm, step i) is the computationally expensive one, even though we have reduced the cost in comparison with Algorithm 1 by only computing the vertices of the previous estimate and using set operations for the remaining sets. Also notice that step iii) reduces the size (values n_c and n_g in Definition 4) of the representation of the constrained zonotope associated with the set-valued estimate. In the literature for LTVs using zonotopes or constrained zonotopes, this is typically included as an additional method to be performed after finding the estimate [5][7]. Therefore, step iii) precludes the need to any of those methods.

3.3. Event-triggering between convex and nonconvex method

One of the main uses of set-membership approaches is to perform fault detection and isolation. In such case, one can consider multiple LPV models as in (1) where one corresponds to the fault-free case and an additional one for each combination of considered faults. Then, detecting and isolating the fault requires to perform model invalidation whenever the produced set for a particular case produces the empty set, i.e., there are no possible values of all the exogenous signals and initial conditions that justify that particular model. Under such scenarios, the question is not to produce the set of all possible state values at time k but rather to check if the set is empty. If the faults can be caused by an intelligent opponent trying to attack the system, accuracy is a vital aspect since added conservatism means additional attacking signals going undetected. However, as seen in Section 2, finding any feasible point to the problem in (2) takes considerable time even for small values of k.

The proposed method in this section is to have an event-triggered mechanism following the idea of incorporating these rules in the context of set-valued estimators [15]. Whenever the elapsed time to solve the feasibility (2) is greater than a given threshold (dependent on how much time the detector has to produce an output regarding the existence of faults), a trigger is generated. Assume that the sequence of triggers is given at times $\tau_0, \tau_1, \cdots$ with $\tau_0 = 0$. At time τ_1, the detector will do the following procedure:

i) Compute the set-valued estimates for the current time τ_1, $X(\tau_1)$ using the exact convex hull method from the set $X(\tau_0) = X(0)$;

i) Replace in (2) the condition $\mathbf{x}(0) \in X(0)$ by $\mathbf{x}(0) \in X(\tau_1)$;

i) The last constraint should use the measurements $y(\tau_1+1), y(\tau_1+2), \cdots$ instead of $y(\tau_0+1), y(\tau_0+2), \cdots$;

i) Repeat for all events $\tau_2, \tau_3, \cdots$.

The above procedure is solving the computationally hard feasibility problem in (2) since the last triggering time τ_j up to the current time instant k. The main advantage is that faults are checked based on the exact nonconvex set (more accurate) at time k from the convex hull set produced at time τ_j. Since triggers happen when the computing time is larger than some constant, the procedure can be run online. However, there is still added conservatism in every event $\tau_1, \tau_2, \cdots$ as the convex hull is computed to replace the known bound for a past state value and reset the number of constraints and optimization variables in (2).

4. Simulations

In this section, simulations are presented in order to illustrate the proposed algorithms (set-based labeled as "CZ approach" and point-based labeled as "ConvexHull of points" for the uncertainties) along with the nonconvex approach for comparison. We consider a motor speed model with state space representation in continuous time given by:

$$\frac{d}{dt}\begin{bmatrix}\dot{\theta}\\ i\end{bmatrix} = \begin{bmatrix}-\frac{b}{J} & \frac{K}{J}\\ -\frac{K}{L} & -\frac{R}{L}\end{bmatrix}\begin{bmatrix}\dot{\theta}\\ i\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{L}\end{bmatrix} V$$

with source voltage V as input and rotational speed of the shaft $\dot{\theta}$ as output, where i is the armature current. We consider the following nominal system constants: moment of inertia of the rotor $J = 0.01$ kg m^2, motor viscous friction constant $b = 0.1$ N m s, K to represent the equal electromotive force constant $Ke = 0.01$ V/rad/s and motor torque constant $Kt = 0.01$ N m/Amp, electric resistance $R = 1$ Ohm and electric inductance $L = 0.5$ H. In the first simulation, it is assumed that the value of b is uncertain and contained in a range $[0.09, 0.011]$. We proceeded to discretize the system using a sampling time of $Ts = 0.1$ s and resorting to the method of zero-order hold on the inputs. During the simulation the system is responding to a unit step as a reference. Both disturbance and noise signals have infinity norm equal to unity and matrices $L = 0.2I$ and $N = 1$.

Figure 4 depicts the evolution of the involved sets for the two main approaches presented in this paper: the approximation algorithm based on set operations and the exact convex method resorting to point-based propagation for the dynamics. For comparison, we present the solution to the nonconvex feasibility problem which stands for the optimal set. Throughout the whole simulation, the optimal set $\Theta(k)$ remained a convex polytope, which meant that the produced sets for the various values of k represented similar results. All three methods produce the same set, as given in Theorem 2 albeit with very distinct computational costs.

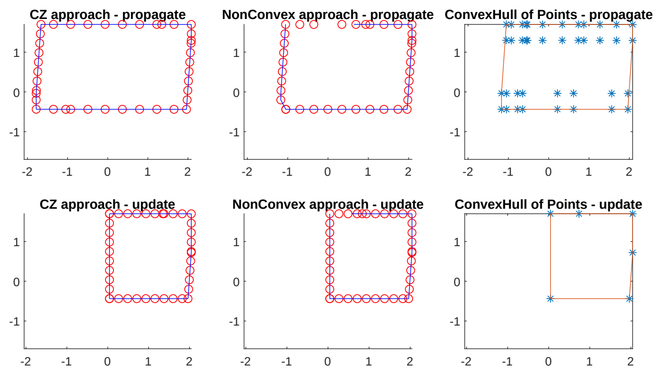

Figure 4: The produced polytopes for the motor speed example at time $k = 5$ using the approximation algorithm (set-based method) for the propagated set (up) and the updated set (bottom) are given on the left. The same sets are given using the nonconvex feasibility approach (middle) and using the exact convex method resorting to point-based operations (right).

In a more challenging simulation, we have considered the moment of inertia of the rotor to be uncertain. In this case, the optimal set is no longer convex given that the bilinear constraint cannot be represented by a rank one uncertainty. In Figure 5, it is depicted the produced sets for $k = 1$. An interesting remark is that the approximation algorithm produces a very conservative set in comparison with the other two approaches. Nevertheless, that difference is less noticeable in the updated sets given the considered bound for the noise. In systems with larger noise sets, the conservatism will be larger and integrated in the propagation step of the algorithm.

At iteration $k = 5$, the propagated set becomes convex. Figure 6 depicts the sets at iteration $k = 6$ and we recover the typical behavior where the approximation is conservative but the exact convex hull of the feasibility set can still be computed by the proposed algorithm with a point-based operation. An important remark is that the computation including the enumeration of the vertices and the final convex hull took at most 0.0439 seconds, meaning that the method can be run as an online state estimator even for smaller sampling times.

An important aspect in set estimation is to determine whether the produced sets are bounded in terms of their hyper-volume. For that reason, we ran the previous simulation for 100 seconds and depict in the following plots the main characteristics regarding the various algorithms.

The sets produced at the final iteration $k = 1000$ are depicted in Figure 7 which shows the relative conservatism of an algorithm based on set operations as opposed to the true convex hull of the nonconvex set. As discussed previously, checking a solution to the nonconvex feasibility problem is prohibitively expensive for the number of variables and constraints used at $k = 1000$.

From Figure 8, we can check that the volume of the set remained bounded throughout the entire simulation. The volume for the non-convex is not pre-

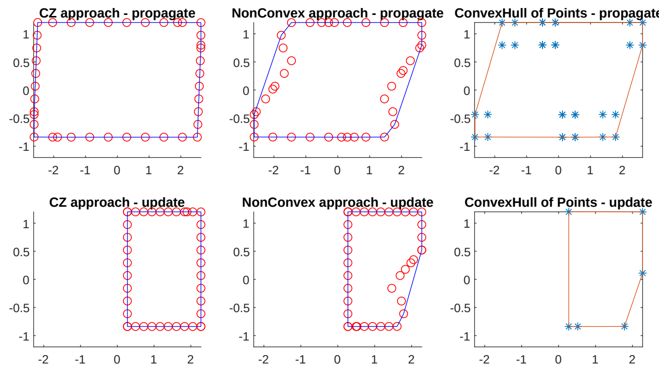

Figure 5: The produced polytopes for the motor speed example at time $k = 1$ using the approximation algorithm (set-based method) for the propagated set (up) and the updated set (bottom) are given on the left. The same sets are given using the nonconvex feasibility approach (middle) and using the exact convex method resorting to point-based operations (right).

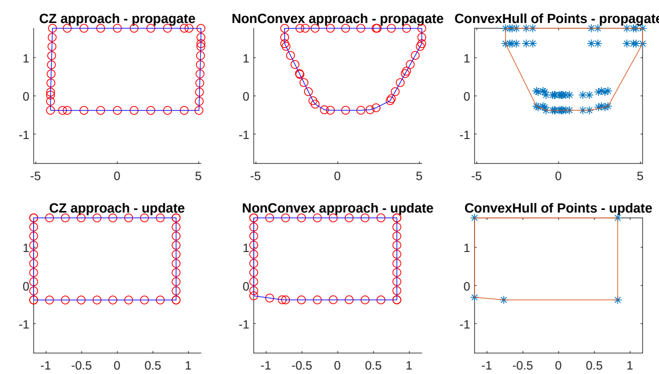

Figure 6: The produced polytopes for the motor speed example at time $k = 6$ using the approximation algorithm (set-based method) for the propagated set (up) and the updated set (bottom) are given on the left. The same sets are given using the nonconvex feasibility approach (middle) and using the exact convex method resorting to point-based operations (right).

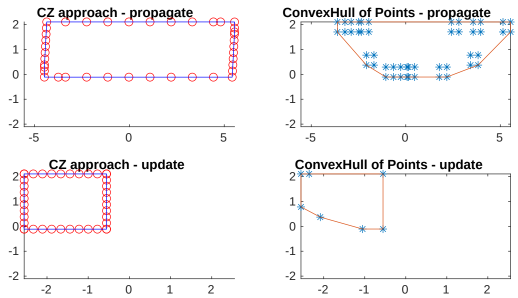

Figure 7: Sets produced by the Constrained Zonotope and Convex Hull of Points algorithms at $k = 1000$ iteration.

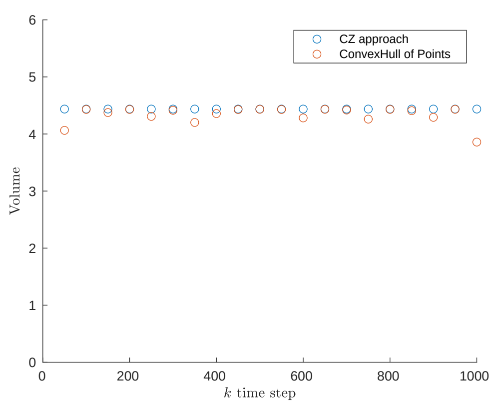

Figure 8: Evolution of the volume of the produced sets every multiple of 50 iterations across the 100 seconds of simulation.

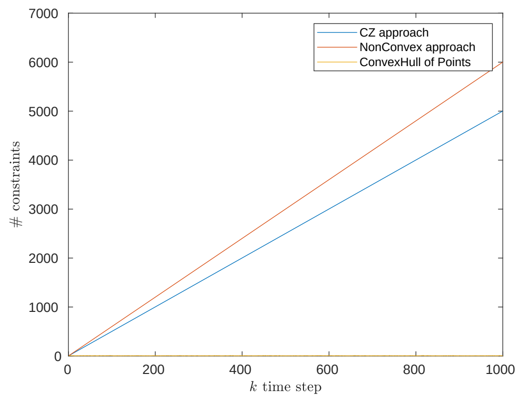

Figure 9: Number of constraints used by each algorithm across the 100 seconds simulation.

sented as, apart from being hard to compute, its description is in the form of the solution of a feasibility program that requires significant computing power even for $k = 10$, evaluating it for a time instant at the end of the simulation would be prohibitively expensive.

Figure 9 depicts the evolution of the number of constraints used to define the sets/feasibility programs in all three approaches. As one might expect, the point-based solutions requires zero constraints while both the other have a linear growth. Another important aspect is the number of variables that are being stored within the set definitions. In Figure 10 is shown how this value evolves for each of the algorithms as time progresses. Interestingly, representing the set as a convex hull of points also means that we can easily check whether some of the points are irrelevant to the description and perform sort of an order reduction just as a byproduct of the algorithm itself. Since we have not implemented a specific order reduction for the Constrained Zonotopes, the number of auxiliary variables used in the definition keeps increasing linearly. Both the number of constraints and variables helps explaining the difference in terms of performance with the maximum computing time for the point-based solution being 0.0089 seconds. This is only achieved because the sets for the uncertainties, disturbances and noise are constant throughout the simulation and the vertex enumeration could be performed a single time off-line before the simulation started.

5. Conclusions and Future Work

In this paper, we have tackled the problem of state estimation for uncertain linear systems in scenarios where there is no information regarding the probabilistic nature of the unknown signals. This results in a worst-case view with the produced set-valued estimates representing all possible values for the state. By formalizing the problem as a feasibility program, the state estimation can

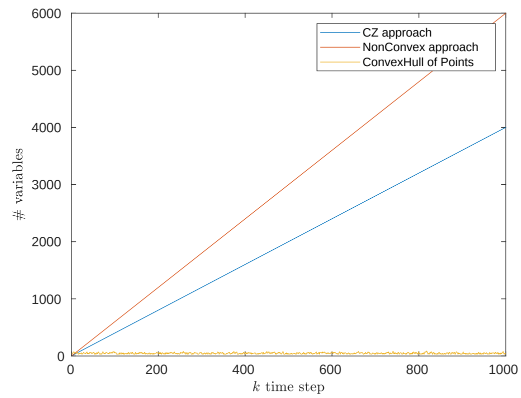

Figure 10: Number of variables/points used by each algorithm across the 100 seconds simulation.

be conducted using a nonconvex solver. It is shown that whenever the solution is a convex set, an algorithm producing the exact set must rely on point-based operations since set-based approaches will be inherently conservative. Exploring this result, we have proposed three methods: i) an approximation algorithm that overbounds the bilinear constraint with a convex one (optimal set-based algorithm); a method to compute the convex hull (optimal convex set) that requires enumerating the vertices of the previous set-valued estimation but employs set operations for the remaining signals; and, iii) an event-triggering algorithm especially useful in fault/attacker detection that uses the nonconvex approach in-between triggers and resets the size of the feasibility program using the method in ii).

The current research opens the possibility to explore three main avenues of future work: i) tackle linear models with uncertain measurement equations; ii) study other practical models for which the estimators can run online; and, iii) investigate conditions under which the optimal solution to the feasibility program is a convex set. Uncertainty in matrix C is harder to incorporate in the approximation algorithm since the measurement set can also be nonconvex, resulting on a research challenge of its own. The topic in ii) would answer one of the harshest criticism of set-membership solutions that are either conservative or do not produce accurate convex sets when applied to uncertain systems with small sampling times. Lastly, understanding the conditions that result in a convex solution would characterize the types of problems for which the point-based method is optimal.

Acknowledgement

This work was partially supported by the Portuguese Fundação para a Ciência e a Tecnologia (FCT) through Institute for Systems and Robotics (ISR), under Laboratory for Robotics and Engineering Systems (LARSyS) project UIDB/50009/2020, through project PCIF/MPG/0156/2019 FirePuma and through COPELABS, University Lusófona project UIDB/04111/2020.

References

[1] Jeff S. Shamma and Michael Athans. Guaranteed properties of gain scheduled control for linear parameter-varying plants. *Automatica*, 27(3):559 – 564, 1991. ISSN 0005-1098. doi: https://doi.org/10.1016/0005-1098(91)90116-J. URL http://www.sciencedirect.com/science/article/pii/000510989190116J.

[2] C. Hoffmann and H. Werner. A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations. *IEEE Transactions on Control Systems Technology*, 23(2):416–433, 2015.

[3] Rihab El Houda Thabet, Tarek Raïssi, Christophe Combastel, Denis Efimov, and Ali Zolghadri. An effective method to interval observer design for time-varying systems. *Automatica*, 50(10):2677 – 2684, 2014. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2014.08.035. URL http://www.sciencedirect.com/science/article/pii/S0005109814003525.

[4] Wentao Tang, Zhenhua Wang, Ye Wang, Tarek Raïssi, and Yi Shen. Interval estimation methods for discrete-time linear time-invariant systems. *IEEE Transactions on Automatic Control*, 64(11):4717–4724, 2019. doi: 10.1109/TAC.2019.2902673.

[5] C. Combastel. A state bounding observer based on zonotopes. In *2003 European Control Conference (ECC)*, pages 2589–2594, 2003.

[6] F.L. Chernousko. Ellipsoidal state estimation for dynamical systems. *Nonlinear Analysis: Theory, Methods & Applications*, 63(5):872 – 879, 2005. ISSN 0362-546X. doi: https://doi.org/10.1016/j.na.2005.01.009. URL http://www.sciencedirect.com/science/article/pii/S0362546X05000210. Invited Talks from the Fourth World Congress of Nonlinear Analysts (WCNA 2004).

[7] Joseph K. Scott, Davide M. Raimondo, Giuseppe Roberto Marseglia, and Richard D. Braatz. Constrained zonotopes: A new tool for set-based estimation and fault detection. *Automatica*, 69:126 – 136, 2016. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2016.02.036. URL http://www.sciencedirect.com/science/article/pii/S0005109816300772.

[8] Daniel Silvestre, Paulo Rosa, João P. Hespanha, and Carlos Silvestre. Stochastic and deterministic fault detection for randomized gossip algorithms. *Automatica*, 78:46 – 60, 2017. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2016.12.011. URL http://www.sciencedirect.com/science/article/pii/S0005109816305192.

[9] Daniel Silvestre. Constrained convex generators: A tool suitable for set-based estimation with range and bearing measurements. *IEEE Control Systems Letters*, 6:1610–1615, 2022. doi: 10.1109/LCSYS.2021.3129729.

[10] Fahed Abdallah, Amadou Gning, and Philippe Bonnifait. Box particle filtering for nonlinear state estimation using interval analysis. *Automatica*, 44(3):807 – 815, 2008. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2007.07.024. URL http://www.sciencedirect.com/science/article/pii/S0005109807003731.

[11] T. Alamo, J.M. Bravo, and E.F. Camacho. Guaranteed state estimation by zonotopes. *Automatica*, 41(6):1035 – 1043, 2005. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2004.12.008. URL http://www.sciencedirect.com/science/article/pii/S0005109805000294.

[12] A. Agung Julius and George J. Pappas. Trajectory based verification using local finite-time invariance. In Rupak Majumdar and Paulo Tabuada, editors, *Hybrid Systems: Computation and Control*, pages 223–236, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. ISBN 978-3-642-00602-9.

[13] B. S. Rego, D. M. Raimondo, and G. V. Raffo. Set-based state estimation of nonlinear systems using constrained zonotopes and interval arithmetic. In *2018 European Control Conference (ECC)*, pages 1584–1589, 2018.

[14] J. Wan, S. Sharma, and R. Sutton. Guaranteed state estimation for nonlinear discrete-time systems via indirectly implemented polytopic set computation. *IEEE Transactions on Automatic Control*, 63(12):4317–4322, 2018.

[15] Daniel Silvestre, Paulo Rosa, João P. Hespanha, and Carlos Silvestre. Self-triggered and event-triggered set-valued observers. *Information Sciences*, 426:61 – 86, 2018. ISSN 0020-0255. doi: https://doi.org/10.1016/j.ins.2017.10.029. URL http://www.sciencedirect.com/science/article/pii/S0020025517310253.

[16] Daniel Silvestre, Paulo Rosa, João P. Hespanha, and Carlos Silvestre. Set-based fault detection and isolation for detectable linear parameter-varying systems. *International Journal of Robust and Nonlinear Control*, 27(18):4381–4397, 2017. doi: 10.1002/rnc.3814. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.3814.

[17] Daniel Silvestre, Paulo Rosa, João P. Hespanha, and Carlos Silvestre. Fault detection for lpv systems using set-valued observers: A coprime factorization approach. *Systems & Control Letters*, 106:32 – 39, 2017. ISSN 0167-6911. doi: https://doi.org/10.1016/j.sysconle.2017.05.007. URL http://www.sciencedirect.com/science/article/pii/S0167691117301019.

[18] D. Silvestre, P. Rosa, J. P. Hespanha, and C. Silvestre. Sensitivity analysis for linear systems based on reachability sets. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 361–366, 2019.

[19] Daniel Silvestre, Paulo Rosa, and Carlos Silvestre. Distinguishability of discrete-time linear systems. *International Journal of Robust and Nonlinear Control*, 31(5):1452–1478, 2021. doi: https://doi.org/10.1002/rnc.5367. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.5367.

[20] J.S. Shamma and Kuang-Yang Tu. Set-valued observers and optimal disturbance rejection. *IEEE Transactions on Automatic Control*, 44(2):253 –264, feb 1999. ISSN 0018-9286.